Dual, 16-Bit, 1230 MSPS, TxDAC+® Digital-to-Analog Converter

AD9122

FEATURES

Flexible LVDS interface allows word, byte, or nibble load
Single-carrier W-CDMA ACLR = 82 dBc @ 122.88 MHz IF
Analog output: adjustable 8.7 mA to 31.7 mA, R_L = 25 Ω to 50 Ω
Novel 2×/4×/8× interpolator/complex modulator allows carrier placement anywhere in the DAC bandwidth
Gain and phase adjustment for sideband suppression
Multiple chip synchronization interfaces
High performance, low noise PLL clock multiplier
Digital inverse sinc filter
Low power: 1.5 W @ 1.2 GSPS, 800 mW @ 500 MSPS, full operating conditions
72-lead, exposed paddle LFCSP

APPLICATIONS

Wireless infrastructure
W-CDMA, CDMA2000, TD-SCDMA, WiMAX, GSM, LTE
Digital high or low IF synthesis
Transmit diversity
Wideband communications: LMDS/MMDS, point-to-point

GENERAL DESCRIPTION

The AD9122 is a dual 16-bit, high dynamic range, digital-to-analog converter (DAC) that provides a sample rate of 1200 MSPS, permitting a multicarrier generation up to the Nyquist frequency. It includes features optimized for direct conversion transmit applications, including complex digital modulation, and gain and offset compensation. The DAC outputs are optimized to interface seamlessly with analog quadrature modulators, such as the ADL537x F-MOD series from Analog Devices, Inc. A 4-wire serial port interface provides for programming/readback of many internal parameters. Full-scale output current can be programmed over a range of 8.7 mA to 31.7 mA. The AD9122 comes in a 72-lead LFCSP.

PRODUCT HIGHLIGHTS

1. Ultralow noise and intermodulation distortion (IMD) enable high quality synthesis of wideband signals from baseband to high intermediate frequencies.
2. A proprietary DAC output switching technique enhances dynamic performance.
3. The current outputs are easily configured for various single-ended or differential circuit topologies.
4. Flexible LVDS digital interface allows the standard 32-wire bus to be reduced to ½ or ¼ of the width.

TYPICAL SIGNAL CHAIN

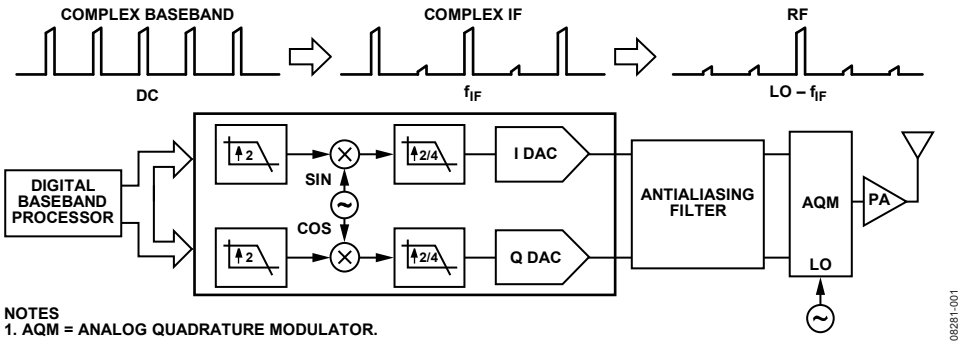

Figure 1.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700 **www.analog.com**
Fax: 781.461.3113

TABLE OF CONTENTS

REVISION HISTORY

3/10—Rev. 0 to Rev. A

Changes to Reflect Differences Between R1 and R2 Silicon ..... Universal
Changes to Features Section 1
Changes to Table 1 5
Changes to Table 2 6
Changes to Table 5 7
Change to IOVDD Rating in Table 6 8
Changes to Table 8 9
Changes to Figure 10 to Figure 15 12
Added Differences Between the AD9122R1 and AD9122R2 Section, Added Figure 36 and Figure 37; Renumbered Sequentially 18
Changes to Table 10 21
Changes to Table 11 23
Changes to FIFO Operation Section 33
Changes to Resettling the FIFO Section and Replaced Table 13; Renumbered Sequentially; Added Serial Port Initiated FIFO Reset Section, and Added FRAME Initiated Relative FIFO Reset Section 34
Added FRAME Initiated Absolute FIFO Reset Section and Replaced Table 14 35
Changes to Figure 54 38
Changes to Table 18 39
Changes to SED Example Section 58
Added Example Start-Up Routine Section 59

9/09—Revision 0: Initial Version

FUNCTIONAL BLOCK DIAGRAM

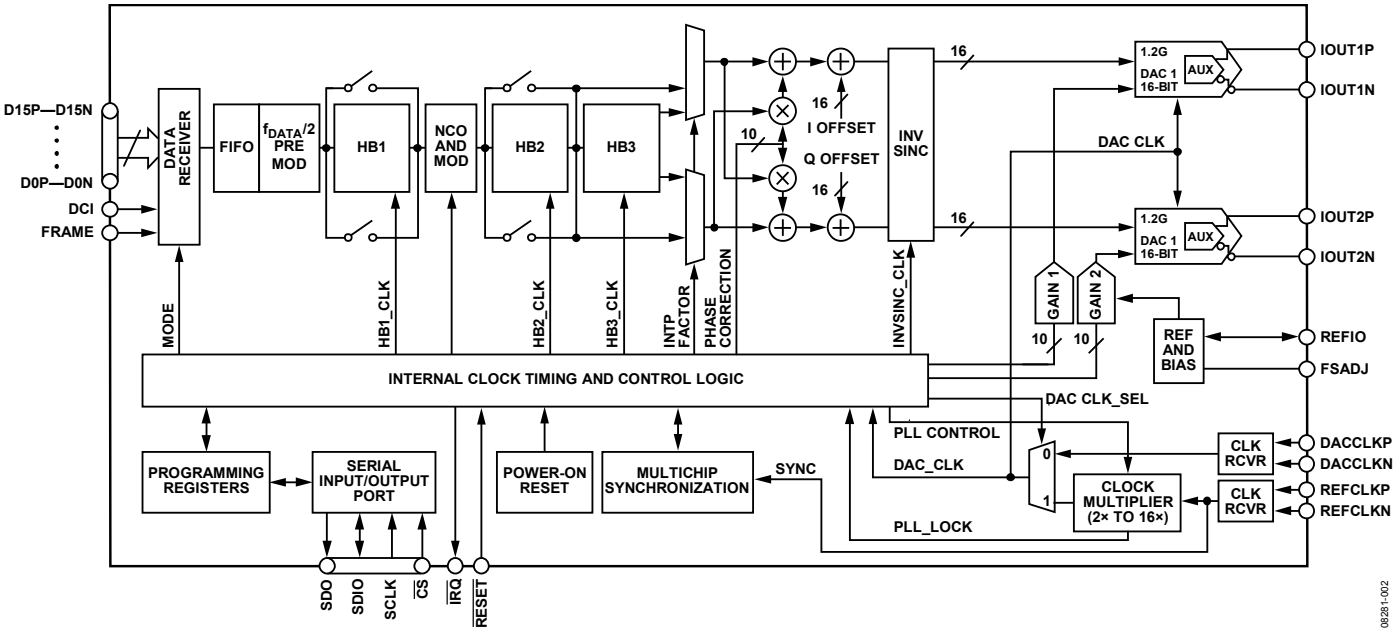

Figure 2. AD9122 Functional Block Diagram

SPECIFICATIONS

DC SPECIFICATIONS

T_{MIN} to T_{MAX}, AVDD33 = 3.3 V, DVDD18 = 1.8 V, CVDD18 =1.8 V, I_{OUTFS} = 20 mA, maximum sample rate, unless otherwise noted.

Table 1.

Parameter	Min	Typ	Max	Unit
RESOLUTION		16		Bits
ACCURACY				
Differential Nonlinearity (DNL)		±2.1		LSB
Integral Nonlinearity (INL)		±3.7		LSB
MAIN DAC OUTPUTS				
Offset Error	−0.001	0	+0.001	% FSR
Gain Error (with Internal Reference)	−3.6	±2	+3.6	% FSR
Full-Scale Output Current[1]	8.66	19.6	31.66	mA
Output Compliance Range	−1.0		+1.0	V
Output Resistance		10		MΩ
Gain DAC Monotonicity		Guaranteed		
Settling Time to Within ±0.5 LSB		20		ns
MAIN DAC TEMPERATURE DRIFT				
Offset		0.04		ppm/°C
Gain		100		ppm/°C
Reference Voltage		30		ppm/°C
REFERENCE				
Internal Reference Voltage		1.2		V
Output Resistance		5		kΩ
ANALOG SUPPLY VOLTAGES				
AVDD33	3.13	3.3	3.47	V
CVDD18	1.71	1.8	1.89	V
DIGITAL SUPPLY VOLTAGES				
DVDD18	1.71	1.8	1.89	V
IOVDD	1.71	1.8/3.3	3.47	V
POWER CONSUMPTION				
2× Mode, f_{DAC} = 491.22 MSPS, IF = 10 MHz, PLL Off		834		mW
2× Mode, f_{DAC} = 491.22 MSPS, IF = 10 MHz, PLL On		913		mW
8× Mode, f_{DAC} = 800 MSPS, IF = 10 MHz, PLL Off		1135	1241	mW
AVDD33		55	57	mA
CVDD18		85	90	mA
DVDD18		444	495	mA
Power-Down Mode (Register 0x01 = 0xF1)		6.5	18.8	mW
Power Supply Rejection Ratio, AVDD33	−0.3		+0.3	% FSR/V
OPERATING RANGE	−40	+25	+85	°C

[1] Based on a 10 kΩ external resistor.

DIGITAL SPECIFICATIONS

T_{MIN} to T_{MAX}, AVDD33 = 1.8 V, IOVDD = 3.3 V, DVDD18 = 1.8 V, CVDD18 = 1.8 V, I_{OUTFS} = 20 mA, maximum sample rate, unless otherwise noted.

Table 2.

Parameter	Conditions	Min	Typ	Max	Unit
CMOS INPUT LOGIC LEVEL					
Input V_{IN} Logic High	IOVDD = 1.8 V	1.2			V
Input V_{IN} Logic High	IOVDD = 2.5 V	1.6			V
Input V_{IN} Logic High	IOVDD = 3.3 V	2.0			V
Input V_{IN} Logic Low	IOVDD = 1.8 V			0.6	V
Input V_{IN} Logic Low	IOVDD = 2.5 V, 3.3 V			0.8	V
CMOS OUTPUT LOGIC LEVEL					
Output V_{OUT} Logic High	IOVDD = 1.8 V	1.4			V
Output V_{OUT} Logic High	IOVDD = 2.5 V	1.8			V
Output V_{OUT} Logic High	IOVDD = 3.3 V	2.4			V
Output V_{OUT} Logic Low	IOVDD = 1.8 V, 2.5 V, 3.3 V			0.4	V
LVDS RECEIVER INPUTS[1]	Applies to DATA, DCI, and FRAME Inputs				
Input Voltage Range, V_{IA} or V_{IB}		825		1675	mV
Input Differential Threshold, V_{IDTH}		−100		+100	mV
Input Differential Hysteresis, V_{IDTHH} to V_{IDTHL}			20		mV
Receiver Differential Input Impedance, R_{IN}		80		120	Ω
LVDS Input Rate	See Table 5				
DAC CLOCK INPUT (DACCLKP, DACCLKN)					
Differential Peak-to-Peak Voltage		100	500	2000	mV
Common-Mode Voltage	Self biased input, ac couple		1.25		V
Maximum Clock Rate		1200			MHz
REFCLK INPUT (REFCLKP, REFCLKN)					
Differential Peak-to-Peak Voltage		100	500	2000	mV
Common-Mode Voltage			1.25		V
REFCLK Frequency (PLL Mode)	1 GHz ≤ f_{VCO} ≤ 2.1 GHz	15.625		600	MHz
REFCLK Frequency (SYNC Mode)	See Multichip Synchronization section for conditions	0		600	MHz
SERIAL PERIPHERAL INTERFACE					
Maximum Clock Rate (SCLK)		40			MHz
Minimum Pulse Width High (t_{PWH})				12.5	ns
Minimum Pulse Width Low (t_{PWOL})				12.5	ns
Setup Time, SDI to SCLK (t_{DS})		1.9			ns
Hold Time, SDI to SCLK (t_{DH})		0.2			ns
Data Valid, SDO to SCLK (t_{DV})		2.3			ns
Setup Time, $\overline{CS}$ to SCLK (t_{DCSB})			1.4		ns

[1] LVDS receiver is compliant to the IEEE 1596 reduced range link, unless otherwise noted.

DIGITAL INPUT DATA TIMING SPECIFICATIONS

Table 3.

Parameter	Min	Typ	Max	Unit
LATENCY (DACCLK Cycles)				
1× Interpolation (With or Without Modulation)		64		Cycles
2× Interpolation (With or Without Modulation)		135		Cycles
4× Interpolation (With or Without Modulation)		292		Cycles
8× Interpolation (With or Without Modulation)		608		Cycles
Inverse Sinc		20		Cycles
Fine Modulation		8		Cycles
Power-Up Time		260		ms

AC SPECIFICATIONS

T_{MIN} to T_{MAX}, AVDD33 = 3.3 V, DVDD18 = 1.8 V, CVDD18 = 1.8 V, I_{OUTFS} = 20 mA, maximum sample rate, unless otherwise noted.

Table 4.

Parameter	Min	Typ	Max	Unit
SPURIOUS-FREE DYNAMIC RANGE (SFDR)				
f_{DAC} = 100 MSPS, f_{OUT} = 20 MHz		78		dBc
f_{DAC} = 200 MSPS, f_{OUT} = 50 MHz		80		dBc
f_{DAC} = 400 MSPS, f_{OUT} = 70 MHz		69		dBc
f_{DAC} = 800 MSPS, f_{OUT} = 70 MHz		72		dBc
TWO-TONE INTERMODULATION DISTORTION (IMD)				
f_{DAC} = 200 MSPS, f_{OUT} = 50 MHz		84		dBc
f_{DAC} = 400 MSPS, f_{OUT} = 60 MHz		86		dBc
f_{DAC} = 400 MSPS, f_{OUT} = 80 MHz		84		dBc
f_{DAC} = 800 MSPS, f_{OUT} = 100 MHz		81		dBc
NOISE SPECTRAL DENSITY (NSD) EIGHT-TONE, 500 kHz TONE SPACING				
f_{DAC} = 200 MSPS, f_{OUT} = 80 MHz		−162		dBm/Hz
f_{DAC} = 400 MSPS, f_{OUT} = 80 MHz		−163		dBm/Hz
f_{DAC} = 800 MSPS, f_{OUT} = 80 MHz		−164		dBm/Hz
W-CDMA ADJACENT CHANNEL LEAKAGE RATIO (ACLR), SINGLE CARRIER				
f_{DAC} = 491.52 MSPS, f_{OUT} = 10 MHz		84		dBc
f_{DAC} = 491.52 MSPS, f_{OUT} = 122.88 MHz		82		dBc
f_{DAC} = 983.04 MSPS, f_{OUT} = 122.88 MHz		83		dBc
W-CDMA SECOND ACLR, SINGLE CARRIER				
f_{DAC} = 491.52 MSPS, f_{OUT} = 10 MHz		88		dBc
f_{DAC} = 491.52 MSPS, f_{OUT} = 122.88 MHz		86		dBc
f_{DAC} = 983.04 MSPS, f_{OUT} = 122.88 MHz		88		dBc

Table 5. Interface Speeds

		f_{BUS} (Mbps)		
Bus Width	**Interpolation Factor**	**1.8 V ± 5%**	**1.8 V ± 2%**	**1.9 V ± 5%**
Nibble (4 Bits)	1×	1100	1200	1230
	2× (HB1)	1100	1200	1230
	2× (HB2)	1100	1200	1230
	4×	1100	1200	1230
	8×	1100	1200	1230
Byte (8 Bits)	1×	1100	1200	1230
	2× (HB1)	1100	1200	1230
	2× (HB2)	1100	1200	1230
	4×	1100	1200	1230
	8×	550	600	615
Word (16 Bits)	1×	1100	1200	1230
	2× (HB1)	900	1000	1000
	2× (HB2)	1100	1200	1230
	4×	550	600	615
	8×	275	300	307.5

ABSOLUTE MAXIMUM RATINGS

Table 6.

Parameter	With Respect To	Rating
AVDD33	AVSS, EPAD, CVSS, DVSS	−0.3 V to +3.6 V
IOVDD	AVSS, EPAD, CVSS, DVSS	−0.3 V to +3.6 V
DVDD18, CVDD18	AVSS, EPAD, CVSS, DVSS	−0.3 V to +2.1 V
AVSS	EPAD, CVSS, DVSS	−0.3 V to +0.3 V
EPAD	AVSS, CVSS, DVSS	−0.3 V to +0.3 V
CVSS	AVSS, EPAD, DVSS	−0.3 V to +0.3 V
DVSS	AVSS, EPAD, CVSS	−0.3 V to +0.3 V
FSADJ, REFIO, IOUT1P/IOUT1N, IOUT2P/IOUT2N	AVSS	−0.3 V to AVDD33 + 0.3 V
D[15:0]P/D[15:0]N, FRAMEP/FRAMEN, DCIP/DCIN	EPAD, DVSS	−0.3 V to DVDD18 + 0.3 V
DACCLKP/DACCLKN, REFCLKP/REFCLKN	DVSS	−0.3 V to CVDD18 + 0.3 V
$\overline{\text{RESET}}$, $\overline{\text{IRQ}}$, $\overline{\text{CS}}$, SCLK, SDIO, SDO	EPAD, DVSS	−0.3 V to IOVDD + 0.3 V
Junction Temperature		125°C
Storage Temperature Range		−65°C to +150°C

Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational section of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

THERMAL RESISTANCE

The exposed paddle (EPAD) must be soldered to the ground plane for the 72-lead, LFCSP. The EPAD performs as an electrical and thermal connection to the board.

Typical θ_{JA}, θ_{JB}, and θ_{JC} are specified for a 4-layer board in still air. Airflow increases heat dissipation effectively reducing θ_{JA} and θ_{JB}.

Table 7. Thermal Resistance

Package	θ_{JA}	θ_{JB}	θ_{JC}	Unit	Conditions
72-Lead LFCSP_VQ	20.7	10.9	1.1	°C/W	EPAD soldered

ESD CAUTION

ESD (electrostatic discharge) sensitive device. Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

PIN CONFIGURATION AND FUNCTION DESCRIPTIONS

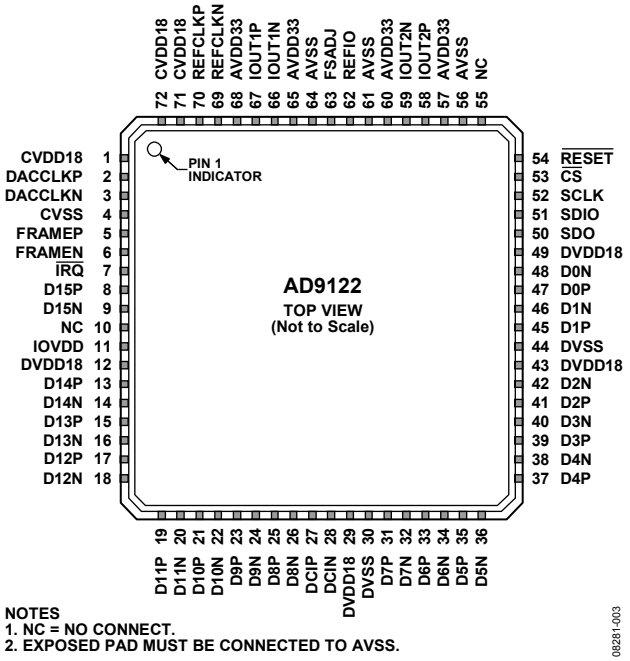

Figure 3. Pin Configuration

Table 8. Pin Function Descriptions

Pin No.	Mnemonic	Description
1	CVDD18	1.8 V Clock Supply. Supplies clock receivers, clock distribution, and PLL circuitry.
2	DACCLKP	DAC Clock Input, Positive.
3	DACCLKN	DAC Clock Input, Negative.
4	CVSS	Clock Supply Common.
5	FRAMEP	Frame Input, Positive.
6	FRAMEN	Frame Input, Negative.
7	$\overline{\text{IRQ}}$	Interrupt Request. Open-drain, active low output. Connect external pull-up to IOVDD.
8	D15P	Data Bit 15 (MSB), Positive.
9	D15N	Data Bit 15 (MSB), Negative.
10	NC	No connection to device.
11	IOVDD	Supply Pin for Serial Ports, $\overline{\text{RESET}}$ and $\overline{\text{IRQ}}$. 1.8 V to 3.3 V can be supplied to this pin.
12	DVDD18	1.8 V Digital Supply. Supplies power to digital core and digital data ports.
13	D14P	Data Bit 14, Positive.
14	D14N	Data Bit 14, Negative.
15	D13P	Data Bit 13, Positive.
16	D13N	Data Bit 13, Negative.
17	D12P	Data Bit 12, Positive.
18	D12N	Data Bit 12, Negative.
19	D11P	Data Bit 11, Positive.
20	D11N	Data Bit 11, Negative.
21	D10P	Data Bit 10, Positive.
22	D10N	Data Bit 10, Negative.
23	D9P	Data Bit 9, Positive.
24	D9N	Data Bit 9, Negative.
25	D8P	Data Bit 8, Positive.

Pin No.	Mnemonic	Description
26	D8N	Data Bit 8, Negative.
27	DCIP	Data Clock Input, Positive.
28	DCIN	Data Clock Input, Negative.
29	DVDD18	1.8 V Digital Supply.
30	DVSS	Digital Common.
31	D7P	Data Bit 7, Positive.
32	D7N	Data Bit 7, Negative.
33	D6P	Data Bit 6, Positive.
34	D6N	Data Bit 6, Negative.
35	D5P	Data Bit 5, Positive.
36	D5N	Data Bit 5, Negative.
37	D4P	Data Bit 4, Positive.
38	D4N	Data Bit 4, Negative.
39	D3P	Data Bit 3, Positive.
40	D3N	Data Bit 3, Negative.
41	D2P	Data Bit 2, Positive.
42	D2N	Data Bit 2, Negative.
43	DVDD18	1.8 V Digital Supply.
44	DVSS	Digital Common.
45	D1P	Data Bit 1, Positive.
46	D1N	Data Bit 1, Negative.
47	D0P	Data Bit 0, Positive.
48	D0N	Data Bit 0, Negative.
49	DVDD18	1.8 V Digital Supply.
50	SDO	Serial Port Data Output (CMOS Levels with Respect to IOVDD).
51	SDIO	Serial Port Data Input/Output (CMOS Levels with Respect to IOVDD).
52	SCLK	Serial Port Clock Input (CMOS Levels With Respect to IOVDD).
53	$\overline{CS}$	Serial Port Chip Select. Active Low (CMOS Levels With Respect to IOVDD).
54	$\overline{RESET}$	Reset. Active Low (CMOS Levels With Respect to IOVDD).
55	NC	No connection to device.
56	AVSS	Analog Supply Common.
57	AVDD33	3.3 V Analog Supply.
58	IOUT2P	Q DAC Positive Current Output.
59	IOUT2N	Q DAC Negative Current Output.
60	AVDD33	3.3 V Analog Supply.
61	AVSS	Analog Supply Common.
62	REFIO	Voltage Reference. Nominally 1.2 V output. Should be decoupled to analog common.
63	FSADJ	Full-Scale Current Output Adjust. Place a 10 kΩ resistor on the analog common.
64	AVSS	Analog Common.
65	AVDD33	3.3 V Analog Supply.
66	IOUT1N	I DAC Negative Current Output.
67	IOUT1P	I DAC Positive Current Output.
68	AVDD33	3.3 V Analog Supply.
69	REFCLKN	PLL Reference Clock Input, Negative. This pin has secondary function as SYNC input.
70	REFCLKP	PLL Reference Clock Input, Positive. This pin has secondary function as SYNC input.
71	CVDD18	1.8 V Clock Supply. Supplies clock receivers, clock distribution, and PLL circuitry.
72	CVDD18	1.8 V Clock Supply. Supplies clock receivers, clock distribution, and PLL circuitry.
	EPAD	Exposed pad must be connected to AVSS. This provides an electrical, thermal, and mechanical connection to the PCB.

TYPICAL PERFORMANCE CHARACTERISTICS

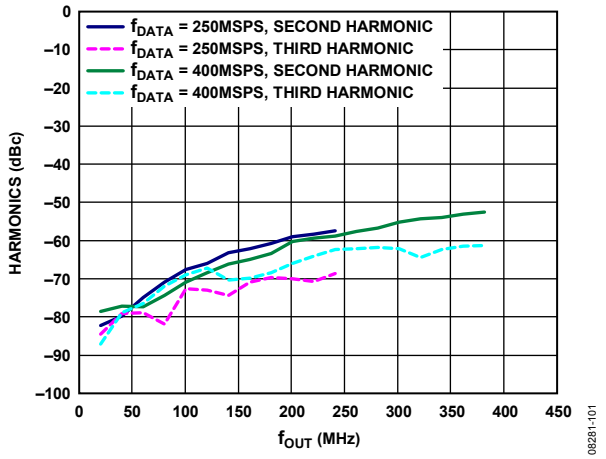

Figure 4. Harmonics vs. f_{OUT} over f_{DATA}, 2× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

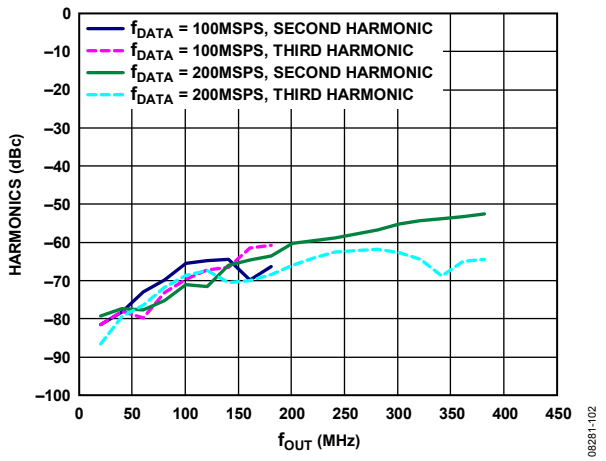

Figure 5. Harmonics vs. f_{OUT} over f_{DATA}, 4× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

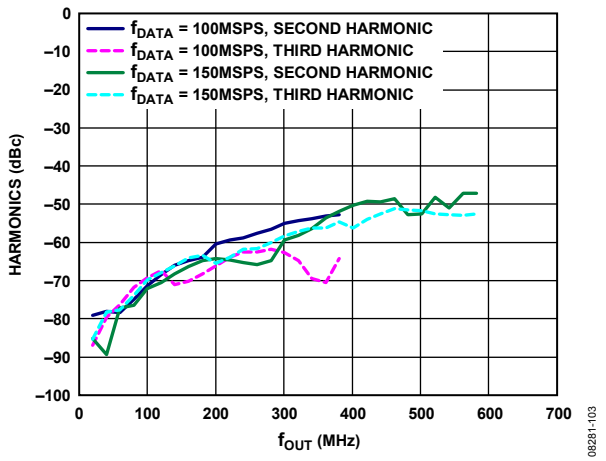

Figure 6. Harmonics vs. f_{OUT} over f_{DATA}, 8× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

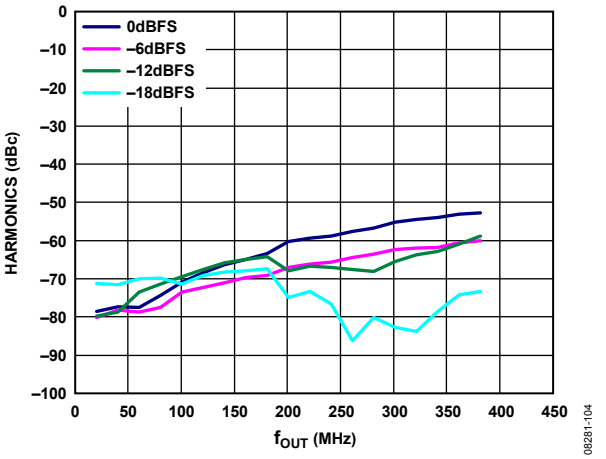

Figure 7. Second Harmonic vs. f_{OUT} over Digital Scale, 2× Interpolation, f_{DATA} = 400 MSPS, f_{SC} = 20 mA

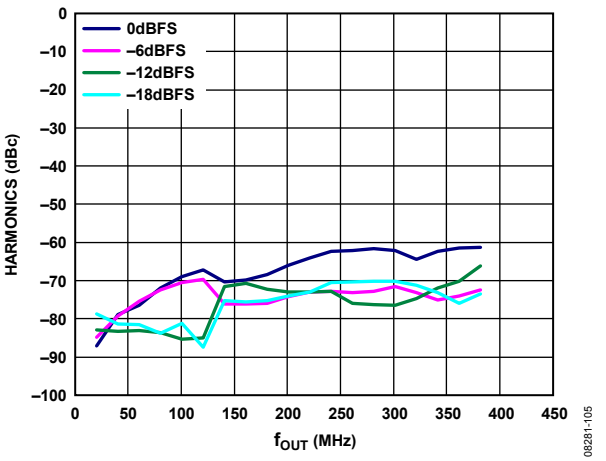

Figure 8. Third Harmonic vs. f_{OUT} over Digital Scale, 2× Interpolation, f_{DATA} = 400 MSPS, f_{SC} = 20 mA

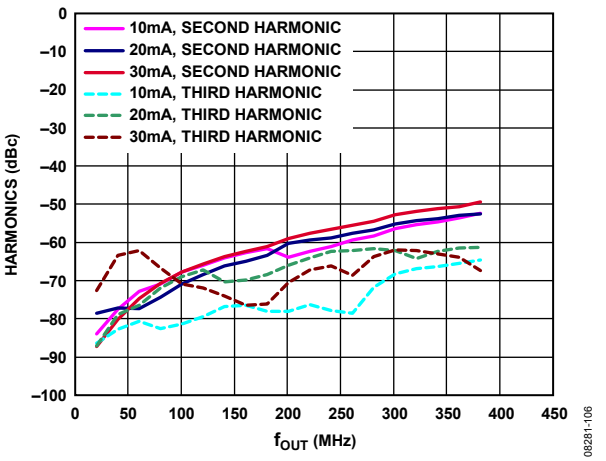

Figure 9. Second Harmonic vs. f_{OUT} over f_{SC}, 2× Interpolation, f_{DATA} = 400 MSPS, Digital Scale = 0 dBFS

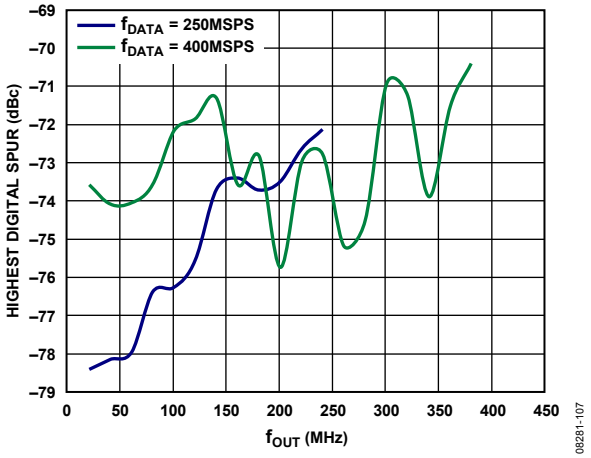

Figure 10. Highest Digital Spur vs. f_{OUT} over f_{DATA}, 2× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

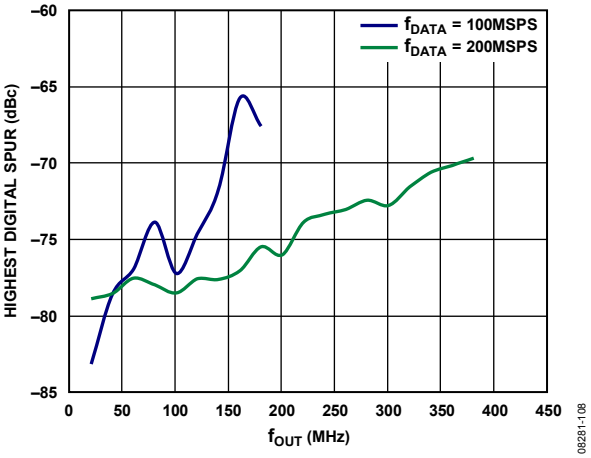

Figure 11. Highest Digital Spur vs. f_{OUT} over f_{DATA}, 4× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

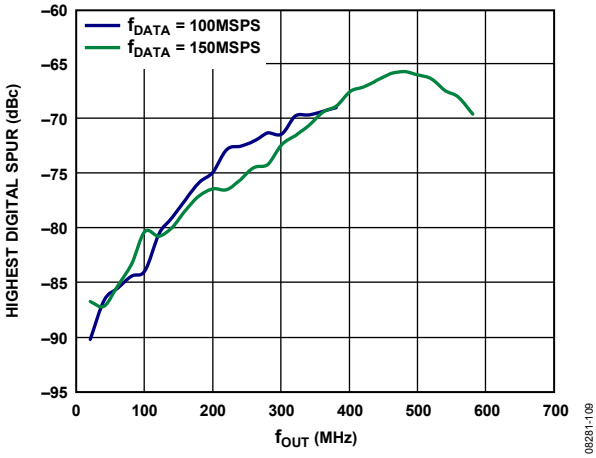

Figure 12. Highest Digital Spur vs. f_{OUT} over f_{DATA}, 8× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

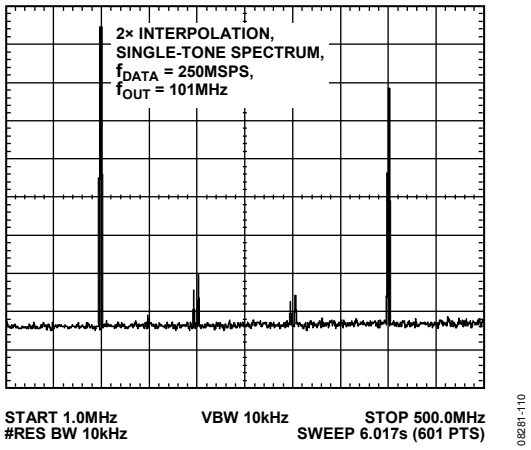

Figure 13. 2× Interpolation, Single-Tone Spectrum, f_{DATA} = 250 MSPS, f_{OUT} = 101 MHz

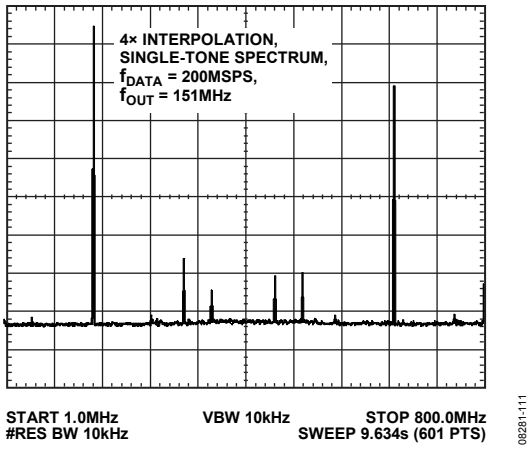

Figure 14. 4× Interpolation, Single-Tone Spectrum, f_{DATA} = 200 MSPS, f_{OUT} = 151 MHz

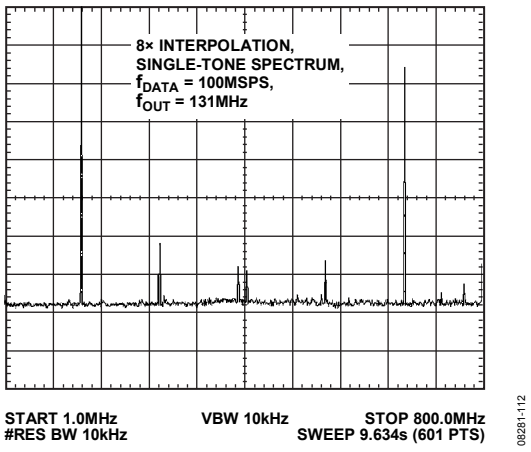

Figure 15. 8× Interpolation, Single-Tone Spectrum, f_{DATA} = 100 MSPS, f_{OUT} = 131 MHz

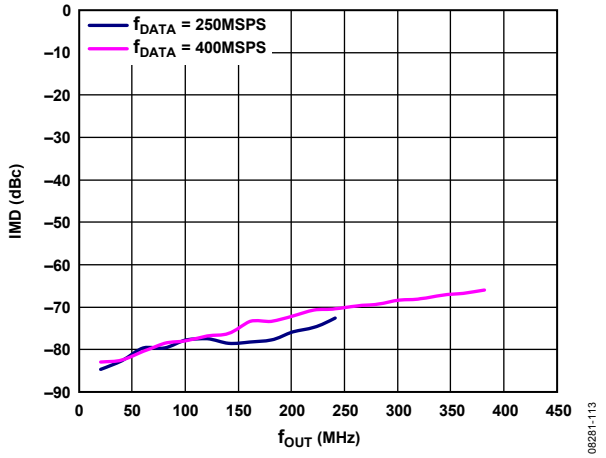

Figure 16. IMD vs. f_{OUT} over f_{DATA}, 2× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

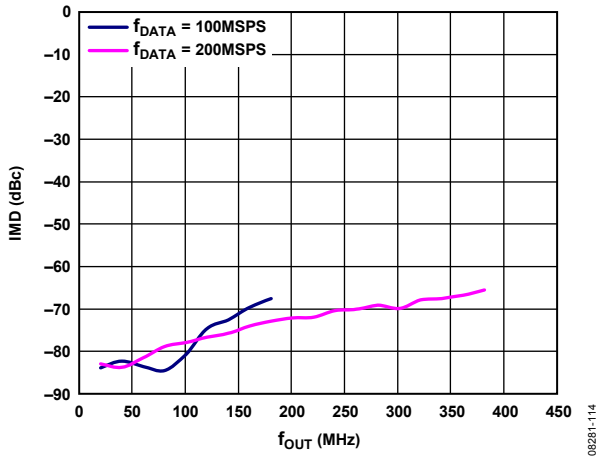

Figure 17. IMD vs. f_{OUT} over f_{DATA}, 4× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

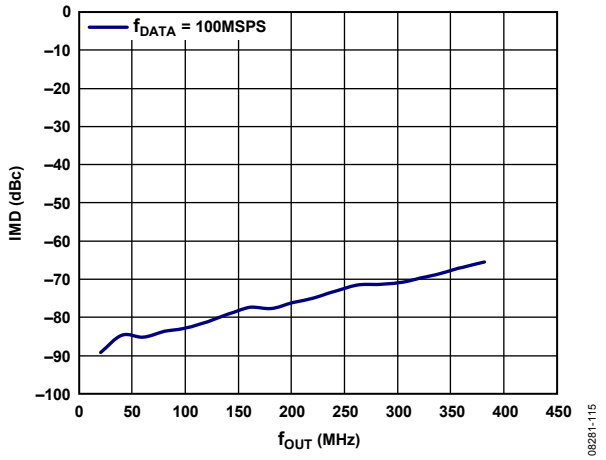

Figure 18. IMD vs. f_{OUT} over f_{DATA}, 8× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

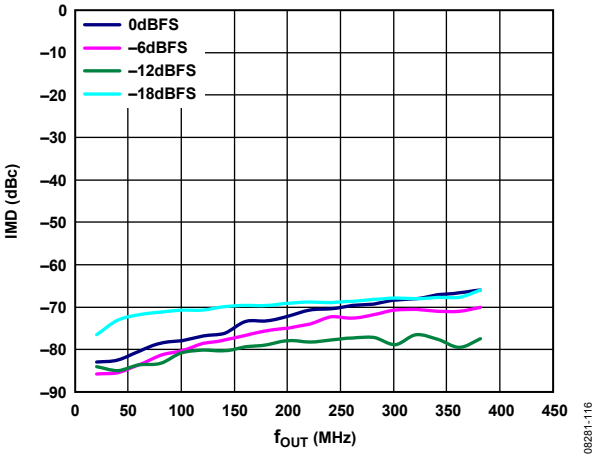

Figure 19. IMD vs. f_{OUT} over Digital Scale, 2× Interpolation, f_{DATA} = 400 MSPS, f_{SC} = 20 mA

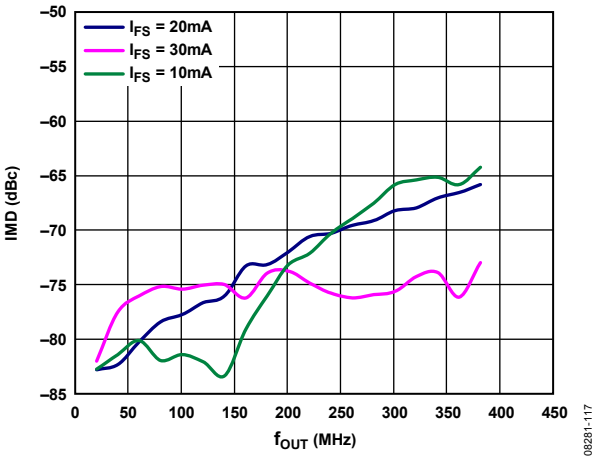

Figure 20. IMD vs. f_{OUT} over f_{SC}, 2× Interpolation, f_{DATA} = 400 MSPS, Digital Scale = 0 dBFS

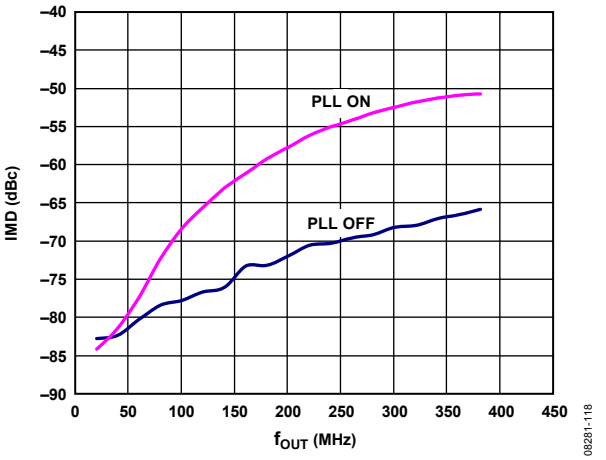

Figure 21. IMD vs. f_{OUT}, PLL On vs. PLL Off, 4× Interpolation, f_{DATA} = 200 MSPS, Digital Scale = 0 dBFS, f_{SC} = 20 mA

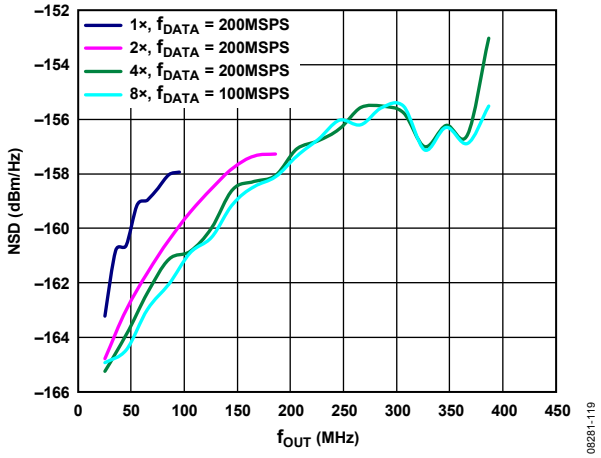

Figure 22. 1-Tone NSD vs. f_{OUT} over Interpolation Rate, Digital Scale = 0 dBFS, f_{SC} = 20 mA, PLL Off

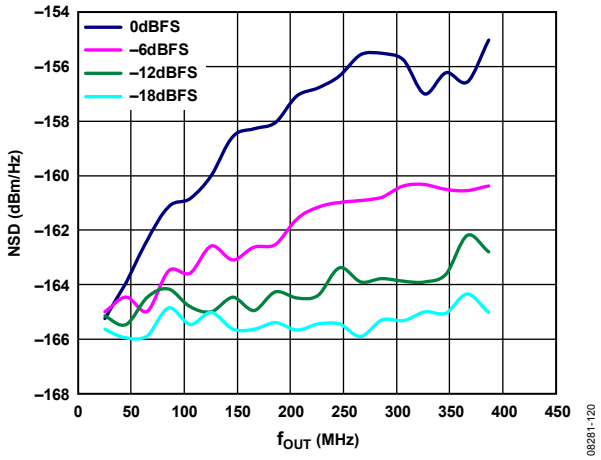

Figure 23. 1-Tone NSD vs. f_{OUT} over Digital Scale, f_{DATA} = 200 MSPS, 4× Interpolation, f_{SC} = 20 mA, PLL Off

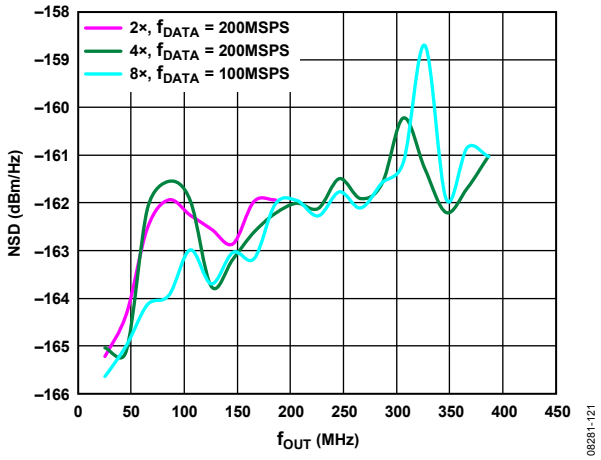

Figure 24. 1-Tone NSD vs. f_{OUT} over Interpolation Rate, Digital Scale = 0 dBFS, f_{SC} = 20 mA, PLL On

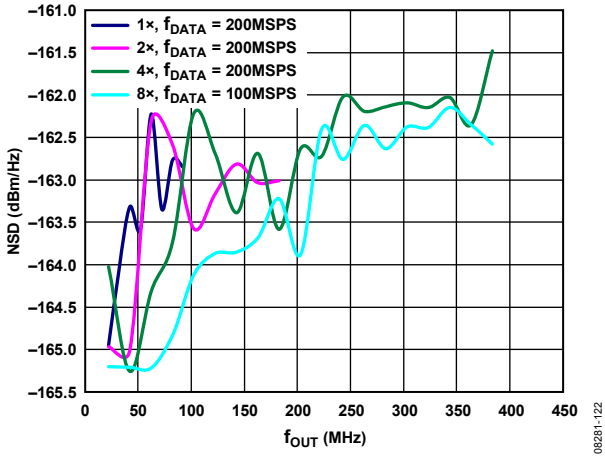

Figure 25. 8-Tone NSD vs. f_{OUT} over Interpolation Rate, Digital Scale = 0 dBFS, f_{SC} = 20 mA, PLL Off

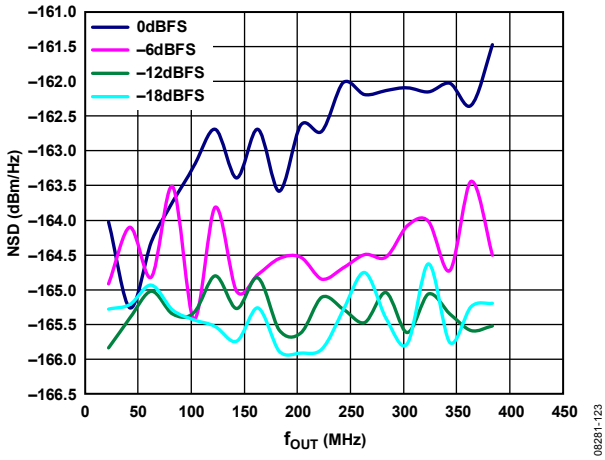

Figure 26. 8-Tone NSD vs. f_{OUT} over Digital Scale, f_{DATA} = 200 MSPS, 4× Interpolation, f_{SC} = 20 mA, PLL Off

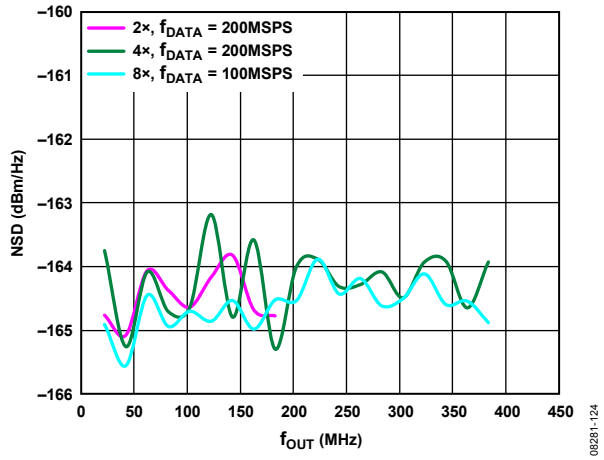

Figure 27. 8-Tone NSD vs. f_{OUT} over Interpolation Rate, Digital Scale = 0 dBFS, f_{SC} = 20 mA, PLL On

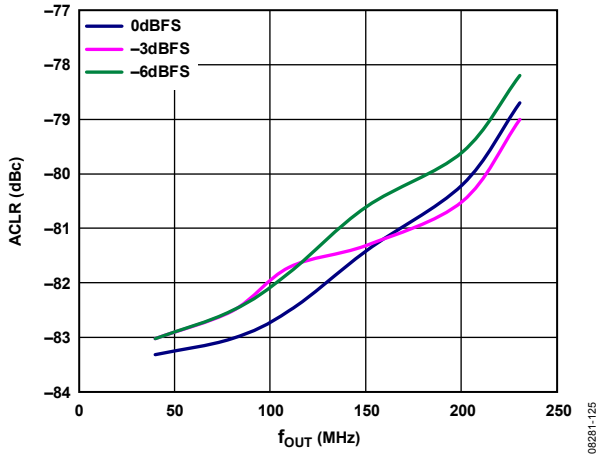

Figure 28. 1-Carrier W-CDMA ACLR vs. f_{OUT} over Digital Scale, Adjacent Channel, PLL Off

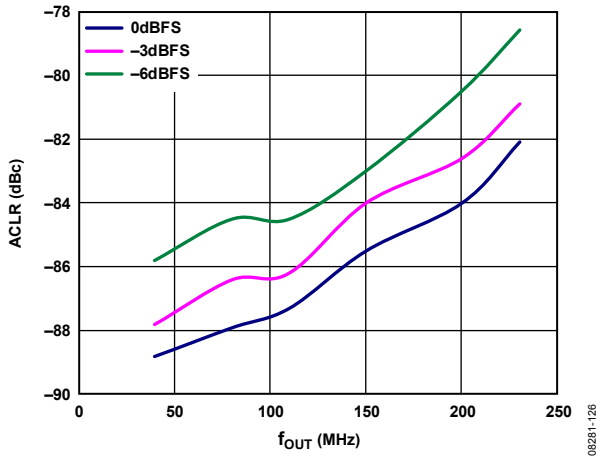

Figure 29. 1-Carrier W-CDMA ACLR vs. f_{OUT} over f_{DAC}, Alternate Channel, PLL Off

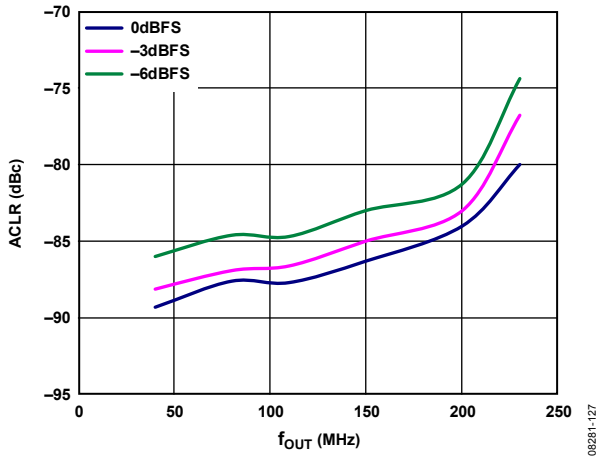

Figure 30. 1-Carrier W-CDMA ACLR vs. f_{OUT} over f_{DAC}, Second Alternate Channel, PLL Off

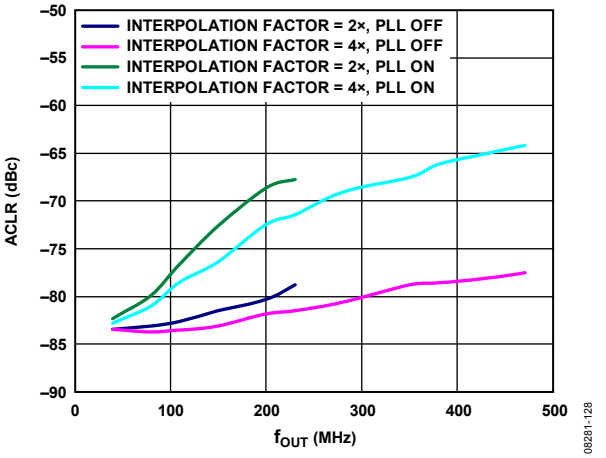

Figure 31. 1-Carrier W-CDMA ACLR vs. f_{OUT}, Adjacent Channel, PLL On vs. PLL Off

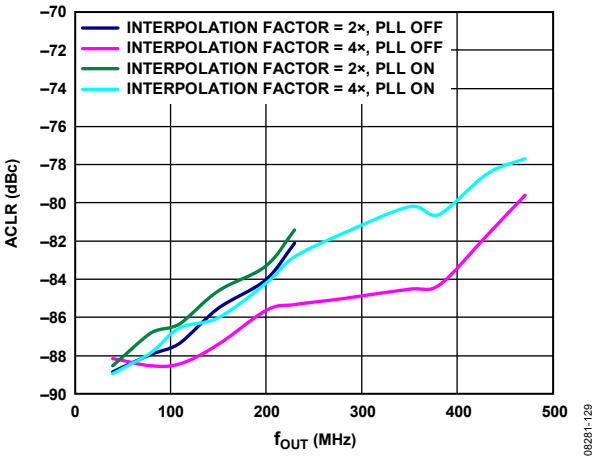

Figure 32. 1-Carrier W-CDMA ACLR vs. f_{OUT}, Alternate Channel, PLL On vs. PLL Off

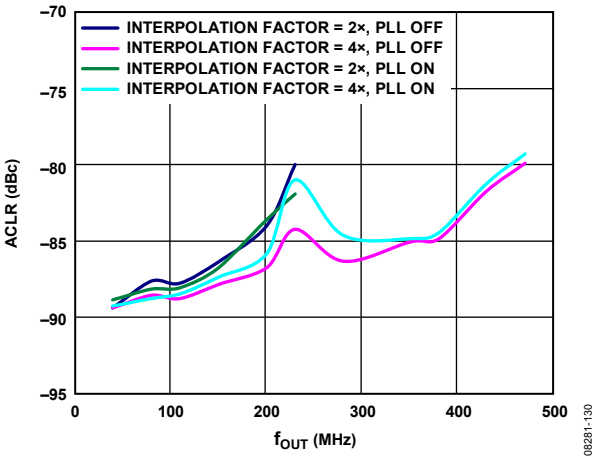

Figure 33. 1-Carrier W-CDMA ACLR vs. f_{OUT}, Second Alternate Channel, PLL On vs. PLL Off

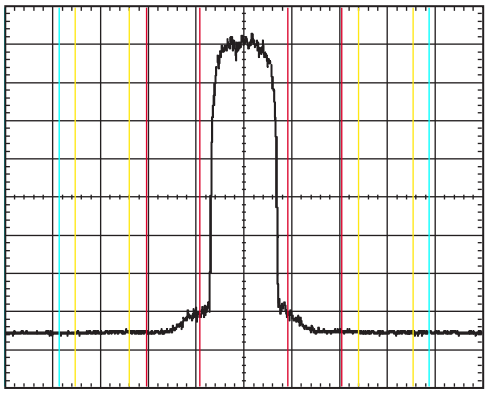

Figure 34. 4-Carrier W-CDMA ACLR Performance, IF = ~150 MHz

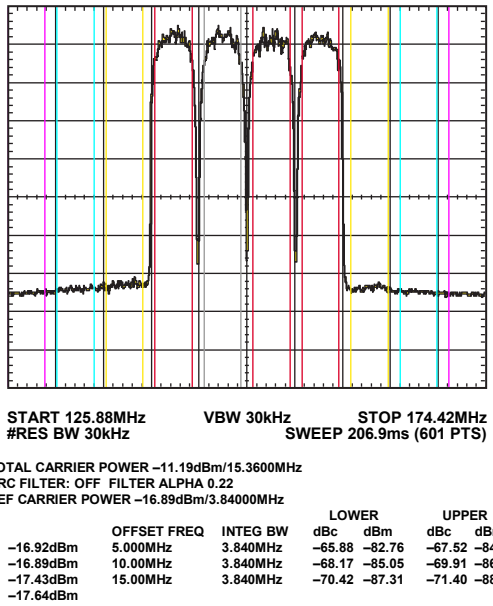

Figure 35. 1-Carrier W-CDMA ACLR Performance, IF = ~150 MHz

TERMINOLOGY

Integral Nonlinearity (INL)
INL is defined as the maximum deviation of the actual analog output from the ideal output, determined by a straight line drawn from zero scale to full scale.

Differential Nonlinearity (DNL)
DNL is the measure of the variation in analog value, normalized to full scale, associated with a 1 LSB change in digital input code.

Offset Error
The deviation of the output current from the ideal of zero is called offset error. For IOUT1P, 0 mA output is expected when the inputs are all 0s. For IOUT1N, 0 mA output is expected when all inputs are set to 1.

Gain Error
The difference between the actual and ideal output span. The actual span is determined by the difference between the output when all inputs are set to 1 and the output when all inputs are set to 0.

Output Compliance Range
The range of allowable voltage at the output of a current output DAC. Operation beyond the maximum compliance limits can cause either output stage saturation or breakdown, resulting in nonlinear performance.

Temperature Drift
Temperature drift is specified as the maximum change from the ambient (25°C) value to the value at either T_{MIN} or T_{MAX}. For offset and gain drift, the drift is reported in ppm of full-scale range (FSR) per degree Celsius. For reference drift, the drift is reported in ppm per degree Celsius.

Power Supply Rejection (PSR)
The maximum change in the full-scale output as the supplies are varied from minimum to maximum specified voltages.

Settling Time
The time required for the output to reach and remain within a specified error band around its final value, measured from the start of the output transition.

Spurious Free Dynamic Range (SFDR)
The difference, in decibels, between the peak amplitude of the output signal and the peak spurious signal within the dc to the Nyquist frequency of the DAC. Typically, energy in this band is rejected by the interpolation filters. This specification, therefore, defines how well the interpolation filters work and the effect of other parasitic coupling paths to the DAC output.

Signal-to-Noise Ratio (SNR)
SNR is the ratio of the rms value of the measured output signal to the rms sum of all other spectral components below the Nyquist frequency, excluding the first six harmonics and dc. The value for SNR is expressed in decibels.

Interpolation Filter
If the digital inputs to the DAC are sampled at a multiple rate of f_{DATA} (interpolation rate), a digital filter can be constructed that has a sharp transition band near $f_{DATA}/2$. Images that typically appear around f_{DAC} (output data rate) can be greatly suppressed.

Adjacent Channel Leakage Ratio (ACLR)
The ratio in decibels relative to the carrier (dBc) between the measured power within a channel relative to its adjacent channel.

Complex Image Rejection
In a traditional two-part upconversion, two images are created around the second IF frequency. These images have the effect of wasting transmitter power and system bandwidth. By placing the real part of a second complex modulator in series with the first complex modulator, either the upper or lower frequency image near the second IF can be rejected.

DIFFERENCES BETWEEN THE AD9122R1 AND AD9122R2

The AD9122 underwent a die revision in early 2010, that incremented the die revision from R1 to R2. The following list explains the differences between the revisions.

- IOVDD supply voltage range.
  For the AD9122R1, the valid operational range for IOVDD is 1.8 V to 2.5 V ± 10%. For the AD9122R2, the valid operational voltage range is 1.8 V to 3.3 V ± 10%.
- Reduction in spurs level variation.
  The AD9122R1 has a variation of the $f_{DATA} \pm f_{OUT}$ spur between device startups. The AD9122R2 has a consistent and lower $f_{DATA} \pm f_{OUT}$ spur level. (The AD9122R2 still has a spur level variation between power cycles of about 5 dB if PLL is enabled.)
- DCI delay feature added.
  The AD9122R2 has a programmable delay associated with the DCI signal. There are four programmable delay options. The 00 setting gives minimum delay and leaves the timing unchanged from the AD9122R1. Additional delay can be added which may improve timing margins in some systems. The resulting timing options are shown in Table 14.
- Power-down mode power consumption increase.
  The maximum power-down mode power consumption of the R1 devices is 9.8 mW. This power consumption increased to 18.8 mW in the R2 devices.
- Configuration register map changes.
  Register 0x0B, Bit 5:
  AD9122R1→Enable VCO
  AD9122R2→Inactive bit. The VCO is now enabled when the PLL is enabled.
  Register 0x16, Bits[1:0]:
  AD9122R1 → Unused
  AD9122R2 → These bits control the delay of the DCI signal. 00 = minimum delay, 11 = maximum delay.
  Register 0x7F, Bits[5:2]:
  AD9122R1 → Version ID = 0x1
  AD9122R2 → Version ID = 0x2

Device Marking of AD9122 R1 and AD9122 R2

Revision 1 devices are marked as shown in Figure 36. All Revision 1 devices have date codes of earlier than #1021.

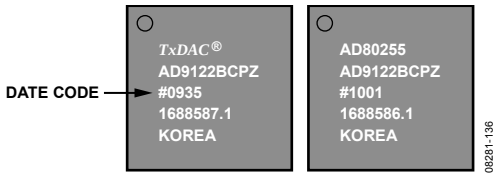

Figure 36. AD9122, Revision 1 Marking

Revision 2 devices are marked as shown in Figure 37. All Revision 2 devices have date codes of #1021 or later.

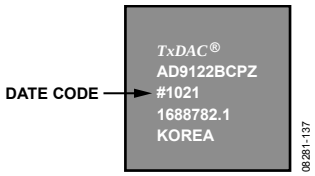

Figure 37. Revision 2 Silicon, AD9122BCPZ Marking

THEORY OF OPERATION

The AD9122 combines many features that make it a very attractive DAC for wired and wireless communications systems. The dual digital signal path and dual DAC structure allow an easy interface to common quadrature modulators when designing single sideband transmitters. The speed and performance of the AD9122 allows wider bandwidths and more carriers to be synthesized than in previously available DACs. In addition, these devices include an innovative low power, 32-bit complex NCO that greatly increases the ease of frequency placement.

The AD9122 offers features that allow simplified synchronization with incoming data and between multiple devices. Auxiliary DACs are also provided on chip for output dc offset compensation (for LO compensation in SSB transmitters) and for gain matching (for image rejection optimization in SSB transmitters).

SERIAL PORT OPERATION

The serial port is a flexible, synchronous serial communications port allowing easy interface to many industry-standard microcontrollers and microprocessors. The serial I/O is compatible with most synchronous transfer formats, including both the Motorola SPI® and Intel® SSR protocols. The interface allows read/write access to all registers that configure the AD9122. Single or multiple byte transfers are supported, as well as MSB-first or LSB-first transfer formats. The serial interface ports can be configured as a single pin I/O (SDIO) or two unidirectional pins for input/output (SDIO/SDO).

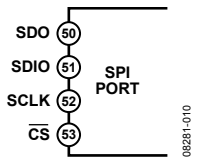

Figure 38. Serial Port Interface Pins

There are two phases to a communication cycle with the AD9122. Phase 1 is the instruction cycle (the writing of an instruction byte into the device), coincident with the first eight SCLK rising edges. The instruction byte provides the serial port controller with information regarding the data transfer cycle, Phase 2 of the communication cycle. The Phase 1 instruction byte defines whether the upcoming data transfer is a read or write and the starting register address for the first byte of the data transfer. The first eight SCLK rising edges of each communication cycle are used to write the instruction byte into the device.

A logic high on the $\overline{CS}$ pin followed by a logic low resets the serial port timing to the initial state of the instruction cycle. From this state, the next eight rising SCLK edges represent the instruction bits of the current I/O operation.

The remaining SCLK edges are for Phase 2 of the communication cycle. Phase 2 is the actual data transfer between the device and the system controller. Phase 2 of the communication cycle is a transfer of one or more data bytes. Registers change immediately upon writing to the last bit of each transfer byte, except for the frequency tuning word and NCO phase offsets that only change when the frequency update bit (Register 0x36, Bit 0) is set.

DATA FORMAT

The instruction byte contains the information shown in Table 9.

Table 9. Serial Port Instruction Byte

I7 (MSB)	I6	I5	I4	I3	I2	I1	I0 (LSB)
R/$\overline{W}$	A6	A5	A4	A3	A2	A1	A0

R/$\overline{W}$, Bit 7 of the instruction byte, determines whether a read or a write data transfer occurs after the instruction byte write. Logic 1 indicates a read operation, and Logic 0 indicates a write operation.

A6 to A0, Bit 6 to Bit 0 of the instruction byte, determine the register that is accessed during the data transfer portion of the communication cycle. For multibyte transfers, A6 is the starting byte address. The remaining register addresses are generated by the device based on the LSB_FIRST bit (Register 0x00, Bit 6).

SERIAL PORT PIN DESCRIPTIONS

Serial Clock (SCLK)

The serial clock pin synchronizes data to and from the device and runs the internal state machines. The maximum frequency of SCLK is 40 MHz. All data input is registered on the rising edge of SCLK. All data is driven out on the falling edge of SCLK.

Chip Select ($\overline{CS}$)

An active low input starts and gates a communication cycle. It allows more than one device to be used on the same serial communications lines. The SDO and SDIO pins go to a high impedance state when this input is high. During the communication cycle, chip select should stay low.

Serial Data I/O (SDIO)

Data is always written into the device on this pin. However, this pin can be used as a bidirectional data line. The configuration of this pin is controlled by Register 0x00, Bit 7. The default is Logic 0, configuring the SDIO pin as unidirectional.

Serial Data Out (SDO)

Data is read from this pin for protocols that use separate lines for transmitting and receiving data. In the case where the device operates in a single bidirectional I/O mode, this pin does not output data and is set to a high impedance state.

SERIAL PORT OPTIONS

The serial port can support both MSB-first and LSB-first data formats. This functionality is controlled by LSB_FIRST (Register 0x00, Bit 6). The default is MSB-first (LSB_FIRST = 0).

When LSB_FIRST = 0 (MSB-first), the instruction and data bit must be written from MSB to LSB. Multibyte data transfers in MSB-first format start with an instruction byte that includes the register address of the most significant data byte. Subsequent data bytes should follow from the high address to low address. In MSB-first mode, the serial port internal byte address generator decrements for each data byte of the multibyte communication cycle.

When LSB_FIRST = 1 (LSB-first), the instruction and data bit must be written from LSB to MSB. Multibyte data transfers in LSB-first format start with an instruction byte that includes the register address of the least significant data byte followed by multiple data bytes. The serial port internal byte address generator increments for each byte of the multibyte communication cycle.

The serial port controller data address decrements from the data address written toward 0x00 for multibyte I/O operations if the MSB-first mode is active. The serial port controller address increments from the data address written toward 0x7F for multibyte I/O operations if the LSB-first mode is active.

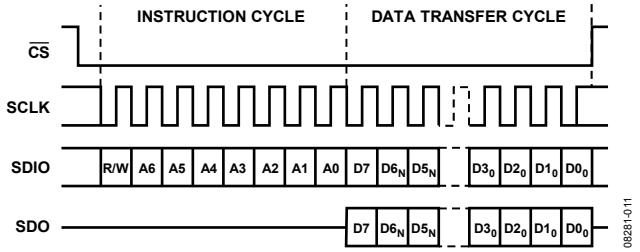

Figure 39. Serial Register Interface Timing MSB-First

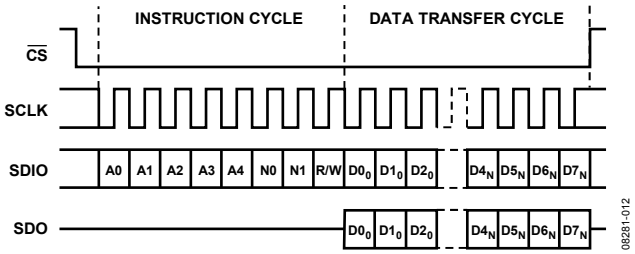

Figure 40. Serial Register Interface Timing LSB-First

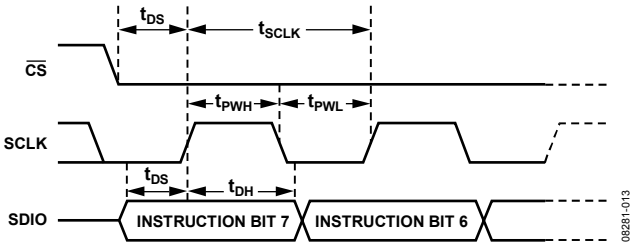

Figure 41. Timing Diagram for Serial Port Register Write

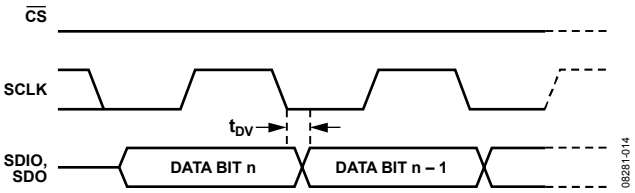

Figure 42. Timing Diagram for Serial Port Register Read

DEVICE CONFIGURATION REGISTER MAP AND DESCRIPTIONS

Table 10. Device Configuration Register Map

Reg Name	Addr (Hex)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Default
Comm	0x00	SDIO	LSB_FIRST	Reset						0x00
Power Control	0x01	Power-down DAC I	Power-down DAC Q	Power-down data receiver	Power-down aux ADC				PLL lock status	0x10
Data Format	0x03	Binary data format	Q data first	MSB swap				Data bus width[1:0]		0x00
Interrupt Enable	0x04	Enable PLL lock lost	Enable PLL locked	Enable sync signal lost	Enable sync signal locked	Enable sync phase locked	Enable soft FIFO sync	Enable FIFO Warning 1	Enable FIFO Warning 2	0x00
Interrupt Enable	0x05	0	0	0	Enable AED compare pass	Enable AED compare fail	Enable SED compare fail	0	0	0x00
Event Flag	0x06	PLL lock lost	PLL locked	Sync signal lost	Sync signal locked	Sync phase locked	Soft FIFO sync	FIFO Warning 1	FIFO Warning 2	N/A
Event Flag	0x07				AED compare pass	AED compare fail	SED compare fail			N/A
Clock Receiver Control	0x08	DACCLK duty correction	REFCLK duty correction	DACCLK cross-correction	REFCLK cross-correction	1	1	1	1	0x3F
PLL Control	0x0A	PLL enable	PLL manual enable	Manual VCO Band[5:0]						0x40
PLL Control	0x0B			PLL VCO enable						0x00
PLL Control	0x0C	PLL Loop Bandwidth[1:0]			PLL Charge Pump Current[4:0]					0xD1
PLL Control	0x0D	N2[1:0]			PLL cross control enable	N0[1:0]		N1[1:0]		0xD9
PLL Status	0x0E	PLL lock				VCO Control Voltage[3:0]				0x00
PLL Status	0x0F			VCO Band Readback[5:0]						0x00
Sync Control	0x10	Sync enable	Data/FIFO rate toggle			Rising edge sync	Sync Averaging[2:0]			0x48
Sync Control	0x11			Sync Phase Request[5:0]						0x00
Sync Status	0x12	Sync lost	Sync locked							N/A
Sync Status	0x13	Sync Phase Readback[7:0] (6.2 format)								N/A
Data Receiver Status	0x15			LVDS FRAME level high	LVDS FRAME level low	LVDS DCI level high	LVDS DCI level low	LVDS data level high	LVDS data level low	N/A
DCI Delay	0x16							DCI Delay[1:0]		0x00
FIFO Control	0x17						FIFO Phase Offset[2:0]			0x04

Reg Name	Addr (Hex)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Default
FIFO Status	0x18	FIFO Warning 1	FIFO Warning 2				FIFO soft align ack	FIFO soft align request	FIFO reset aligned	N/A
FIFO Status	0x19	FIFO Level[7:0]								N/A
Datapath Control	0x1B	Bypass premod	Bypass Sinc^{-1}	Bypass NCO		NCO gain	Bypass phase comp and dc offset	Select sideband	Send I data to Q data	0xE4
HB1 Control	0x1C						HB1[1:0]		Bypass HB1	0x00
HB2 Control	0x1D		HB2[5:0]						Bypass HB2	0x00
HB3 Control	0x1E		HB3[5:0]						Bypass HB3	0x00
Chip ID	0x1F	Chip ID[7:0]								0x08
FTW LSB	0x30	FTW[7:0]								0x00
FTW	0x31	FTW[15:8]								0x00
FTW	0x32	FTW[23:16]								0x00
FTW MSB	0x33	FTW[31:24]								0x08
NCO Phase Offset LSB	0x34	NCO Phase Offset[7:0]								0x00
NCO Phase Offset MSB	0x35	NCO Phase Offset[15:8]								0x00
NCO FTW Update	0x36			Frame FTW ack	Frame FTW request			Update FTW ack	Update FTW request	0x00
I Phase Adj LSB	0x38	I Phase Adj[7:0]								0x00
I Phase Adj MSB	0x39							I Phase Adj[9:8]		0x00
Q Phase Adj LSB	0x3A	Q Phase Adj[7:0]								0x00
Q Phase Adj MSB	0x3B							Q Phase Adj[9:8]		0x00
I DAC Offset LSB	0x3C	I DAC Offset[7:0]								0x00
I DAC Offset MSB	0x3D	I DAC Offset[15:8]								0x00
Q DAC Offset LSB	0x3E	Q DAC Offset[7:0]								0x00
Q DAC Offset MSB	0x3F	Q DAC Offset[15:8]								0x00
I DAC FS Adj.	0x40	I DAC FS Adj[7:0]								0xF9
I DAC Control	0x41	I DAC sleep						I DAC FS Adj[9:8]		0x01
Aux DAC I Data	0x42	I Aux DAC[7:0]								0x00
I Aux DAC Control	0x43	I Aux DAC sign	I Aux DAC current direction	I Aux DAC Sleep				I Aux DAC[9:8]		0x00
Q DAC FS Adj.	0x44	Q DAC FS Adj[7:0]								0xF9
Q DAC Control	0x45	Q DAC sleep						Q DAC FS Adj[9:8]		0x01
Aux DAC Q Data	0x46	Q Aux DAC[7:0]								0x00
Q Aux DAC Control	0x47	Q Aux DAC sign	Q Aux DAC current direction	Q Aux DAC Sleep				Q Aux DAC[9:8]		0x00
Die Temp Range Control	0x48		FS Current[2:0]			Ref Current[2:0]			Capacitor value	0x02
Die Temp LSB	0x49	Die Temp[7:0]								N/A

Reg Name	Addr (Hex)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Default
Die Temp MSB	0x4A	Die Temp[15:8]								N/A
SED Control	0x67	SED compare enable		Sample error detected		Autoclear enable		Compare fail	Compare pass	0x00
Compare I0 LSBs	0x68	Compare Value I0[7:0]								0xB6
Compare I0 MSBs	0x69	Compare Value I0[15:8]								0x7A
Compare Q0 LSBs	0x6A	Compare Value Q0[7:0]								0x45
Compare Q0 MSBs	0x6B	Compare Value Q0[15:8]								0xEA
Compare I1 LSBs	0x6C	Compare Value I1[7:0]								0x16
Compare I1 MSBs	0x6D	Compare Value I1[15:8]								0x1A
Compare Q1 LSBs	0x6E	Compare Value Q1[7:0]								0xC6
Compare Q1 MSBs	0x6F	Compare Value Q1[15:8]								0xAA
SED I LSBs	0x70	Errors Detected I_BITS[7:0]								0x00
SED I MSBs	0x71	Errors Detected I_BITS[15:8]								0x00
SED Q LSBs	0x72	Errors Detected Q_BITS[7:0]								0x00
SED Q MSBs	0x73	Errors Detected Q_BITS[15:8]								0x00
Revision	0x7F			Revision[3:0]						N/A

Table 11. Device Configuration Register Descriptions

Reg Name	Addr (Hex)	Bit	Name	Description	Default
Comm	00	7	SDIO	SDIO Operation. 0 = SDIO operates as an input only. 1 = SDIO operates as bidirectional input/output.	0
		6	LSB_FIRST	Serial port communication LSB or MSB first. 0 = MSB first. 1 = LSB first.	0
		5	Reset	The device is held in reset when this bit written high and is held there until the bit is written low.	0
Power Control	01	7	Power-down DAC I	1 = power down DAC I.	0
		6	Power-down DAC Q	1 = power down DAC Q.	0
		5	Power-down data receiver	1 = power down the input data receiver.	0
		4	Power-down auxiliary ADC	1 = power down auxiliary ADC for temperature sensor.	0
		0	PLL lock status	1 = PLL is locked.	0

Reg Name	Addr (Hex)	Bit	Name	Description	Default
Data Format	03	7	Binary data format	0 = input data is in twos complement format. 1 = input data is in binary format.	0
		6	Q data first	Indicates I/Q data pairing on data input. 0 = I data sent to data receiver first. 1 = Q data sent to data receiver first.	0
		5	MSB swap	Swaps the bit order of the data input port. 0 = order of the data bits corresponds to the pin descriptions 1 = bit designations are swapped; most significant bits become the least significant bits	0
		1:0	Data bus width	Data receiver interface mode. 00 = word mode; 16-bit interface bus width. 01 = byte mode; 8-bit interface bus width. 10 = nibble mode; 4-bit interface bus width. 11 = invalid. See the LVDS Input Data Ports section for details on the operation of the different interface modes.	0
Interrupt Enable	04	7	Enable PLL lock lost	1 = enable interrupt for PLL lock lost.	0
		6	Enable PLL locked	1 = enable interrupt for PLL locked.	0
		5	Enable sync signal lost	1 = enable interrupt for sync signal lock lost.	0
		4	Enable sync signal locked	1 = enable interrupt for sync signal locked.	0
		3	Enable sync phase locked	1 = enable interrupt for clock generation ready.	0
		2	Enable soft FIFO sync	1 = enable interrupt for soft FIFO reset.	0
		1	Enable FIFO Warning 1	1 = enable interrupt for FIFO Warning 1.	0
		0	Enable FIFO Warning 2	1 = enable interrupt for FIFO Warning 2.	0
Interrupt Enable	05	7	Set to 0	Set this bit to 0.	0
		6	Set to 0	Set this bit to 0.	0
		5	Set to 0	Set this bit to 0.	0
		4	Enable AED comparison pass	1 = enable interrupt for AED comparison pass.	0
		3	Enable AED comparison fail	1 = enable interrupt for AED comparison fail.	0
		2	Enable SED comparison fail	1 = enable interrupt for SED comparison fail.	0
		1	Set to 0	Set this bit to 0.	0
		0	Set to 0	Set this bit to 0.	0

Reg Name	Addr (Hex)	Bit	Name	Description	Default
Event Flag	06	7	PLL lock lost	1 = indicates that the PLL, which had been previously locked, has unlocked from the reference signal. This is a latched signal.	0
		6	PLL locked	1 = indicates that the PLL has locked to the reference clock input.	0
		5	Sync signal lost	1 = indicates that the sync logic, which had been previously locked, has lost alignment. This is a latched signal.	0
		4	Sync signal locked	1 = indicates that the sync logic did achieve sync alignment. This is indicated when no phase changes were requested for at least a few full averaging cycles.	0
		3	Sync phase locked	1 = indicates that the internal digital clock generation logic is ready. This occurs when internal clocks are present and stable.	0
		2	Soft FIFO sync	1 = indicates that a FIFO reset originating from a serial port-based request has successfully completed. This is a latched signal.	0
		1	FIFO Warning 1	1 = indicates that the difference between the FIFO read and write pointers is 1.	0
		0	FIFO Warning 2	1 = indicates that the difference between the FIFO read and write pointers is 2.	0
			Note that all bit event flags are cleared by writing the respective bit high.		
Event Flag	07	4	AED comparison pass	1 = indicates that the SED logic detected a valid input data pattern compared against the preprogrammed expected values. This is a latched signal.	0
		3	AED comparison fail	1 = indicates that the SED logic detected an invalid input data pattern comparison against the preprogrammed expected values. This is a latched signal that automatically clears when eight valid I/Q data pairs are received.	0
		2	SED comparison fail	1 = indicates that the SED logic detected an invalid input data pattern comparison against the preprogrammed expected values. This is a latched signal.	
			Note that all bit event flags are cleared by writing the respective bit high.		
Clock Receiver Control	08	7	DACCLK duty correction	1 = enables duty-cycle correction on DACCLK input.	0
		6	REFCLK duty correction	1 = enables duty-cycle correction on REFCLK input.	0
		5	DACCLK cross-correction	1 = enables differential crossing correction on the CLK input.	0
		4	REFCLK cross-correction	1 = enables differential crossing correction on the REFCLK input.	0
PLL Control	0A	7	PLL enable	1 = enables the PLL clock multiplier. REFCLK input is used as the PLL reference clock signal.	0
		6	PLL manual enable	Enables the manual selection of the VCO band. 1 = manual mode; the correct VCO band must be determined by the user.	1
		5:0	Manual VCO band	Selects the VCO band to be used.	0
PLL Control	0B	5	PLL VCO enable	This bit is only active for Version 1 devices. For version 2 devices, this bit is inactive. 0 = disables the PLL VCO. 1 = enables the PLL VCO. Set this bit high prior to enabling PLL.	0

Reg Name	Addr (Hex)	Bit	Name	Description	Default
PLL Control	0C	7:6	PLL Loop Bandwidth[1:0]	Selects the PLL loop filter bandwidth. 00 = narrowest bandwidth. 01 = narrow/medium bandwidth. 10 = medium/wide bandwidth. 11 = widest bandwidth.	3
		4:0	PLL Charge Pump Current[4:0]	Sets the nominal PLL charge pump current. 00000 = lowest current setting. 11111 = highest current setting.	10001
PLL Control	0D	7:6	N2[1:0]	PLL control clock divider. It determines the ratio of the DACCLK rate to the PLL controller clock rate. 00 = f_{DACCLK}/f_{PC_CLK} = 2. 01 = f_{DACCLK}/f_{PC_CLK} = 4. 10 = f_{DACCLK}/f_{PC_CLK} = 8. 11 = f_{DACCLK}/f_{PC_CLK} = 16. f_{PC_CLK} must always be less than 75 MHz.	3
		4	PLL cross control enable	Enable PLL cross point controller.	0
		3:2	N0[1:0]	PLL VCO divider. It determines the ratio of the VCO output to the DACCLK frequencies. 00 = f_{VCO}/f_{DACCLK} = 1. 01 = f_{VCO}/f_{DACCLK} = 2. 10 = f_{VCO}/f_{DACCLK} = 4. 11 = f_{VCO}/f_{DACCLK} = 4.	01
		1:0	N1[1:0]	PLL Loop divider. It determines the ratio of the DACCLK to the REFCLK frequencies. 00 = f_{DACCLK}/f_{REFCLK} = 2. 01 = f_{DACCLK}/f_{REFCLK} = 4. 10 = f_{DACCLK}/f_{REFCLK} = 8. 11 = f_{DACCLK}/f_{REFCLK} = 16.	01
PLL Status	0E	7	PLL lock	The PLL generated clock is tracking the REFCLK input signal.	R
		3:0	VCO Control Voltage[3:0]	VCO Control Voltage readback. See Table 25.	R
PLL Status	0F	5:0	VCO Band Readback[5:0]	Indicates the VCO band currently selected.	R
Sync Control	10	7	Sync enable	1 = enables the synchronization logic.	0
		6	Data/FIFO rate toggle	0 = operates the synchronization at the FIFO reset rate. 1 = operates the synchronization at the data rate.	0
		3	Rising edge sync	0 = sync is initiated on the falling edge of the sync input. 1 = sync is initiated on the rising edge of the sync input.	1
		2:0	Sync Averaging[2:0]	Sets the number of input samples that are averaged in determining the sync phase. 000 = 1. 001 = 2. 010 = 4. 011 = 8. 100 = 16. 101 = 32. 110 = 64. 111 = 128.	0

Reg Name	Addr (Hex)	Bit	Name	Description	Default
Sync Control	11	5:0	Sync Phase Request[5:0]	This sets the requested clock phase offset after sync. The offset unit is in DACCLK cycles. 000000 = 0 DACCLK cycles. 000001 = 1 DACCLK cycle. … 111111 = 63 DACCLK cycles. This enables repositioning of the DAC output with respect to the sync input. The offset can also be used to skew the DAC outputs between the synchronized DACs.	0
Sync Status	12	7	Sync lost	1 = indicates that synchronization had been attained but has been lost.	R
		6	Sync locked	1 = indicates that synchronization has been attained.	R
Sync Status	13	7:0	Sync Phase Readback[7:0]	Indicates the averaged sync phase offset (6.2 format). 00000000 = 0.0. 00000001 = 0.25. … 11111110 = 63.50. 11111111 = 63.75. If this value differs from the requested sync phase value, this indicates sync timing errors.	R
Data Receiver Status	15	5	LVDS FRAME level high	One or both of the LVDS FRAME input signals have exceeded 1.7 V.	R
		4	LVDS FRAME level low	One or both of the LVDS FRAME input signals have crossed below 0.7 V.	R
		3	LVDS DCI level high	One or both of the LVDS DCI input signals have exceeded 1.7 V.	R
		2	LVDS DCI level low	One or both of the LVDS DCI input signals have crossed below 0.7 V.	R
		1	LVDS data level high	One or more of the LVDS Dx input signals have exceeded 1.7 V.	R
		0	LVDS data level low	One or both of the LVDS Dx input signals have crossed below 0.7 V.	R
DCI Delay	16	1:0	DCI Delay[1:0]	This option is only available for the Revision 2 silicon. The DCI Delay bits control the delay applied to the DCI signal. This affects the sampling interval of DCI with respect to the DATA inputs. See Table 14. for complete details. 00: 350 pS delay of DCI signal. 01: 590 pS delay of DCI signal. 10: 800 pS delay of DCI signal. 11: 925 pS delay of DCI signal.	0
FIFO Control	17	2:0	FIFO Phase Offset[2:0]	FIFO write pointer phase offset following FIFO reset. 000 = 0. 001 = 1. … 111 = 7. This is the difference between the read pointer and the write pointer values upon FIFO reset. The optimal value is nominally 4.	0
FIFO Status	18	7	FIFO Warning 1	FIFO read and write pointers within ±1.	0
		6	FIFO Warning 2	FIFO read and write pointers within ±2.	0
		2	FIFO soft align acknowledge	FIFO read and write pointers are aligned after serial port initiated FIFO reset.	
		1	FIFO soft align request	Request FIFO read and write pointers alignment via serial port.	0
		0	FIFO reset aligned	FIFO read and write pointers aligned after hardware reset.	0

Reg Name	Addr (Hex)	Bit	Name	Description	Default
FIFO Status	19	7:0	FIFO Level[7:0]	Thermometer encoded measure of the FIFO level.	0
Datapath Control	1B	7	Bypass Premod	1 = bypasses $f_S/2$ premodulator.	1
		6	Bypass Sinc^{-1}	1 = bypasses inverse sinc filter.	1
		5	Bypass NCO	1 = bypasses NCO.	1
		3	NCO gain	0 = default. No gain scaling is applied to the NCO input to the internal digital modulator. 1 = gain scaling of 0.5 is applied to the NCO input to the internal digital modulator. This can eliminate saturation of the modulator output for some combinations of data inputs and NCO signals.	0
		2	Bypass phase compensation and dc offset	1 = bypasses phase compensation.	1
		1	Select sideband	0 = the modulator outputs high-side image. 1 = the modulator outputs low-side image. The image is spectrally inverted compared to the input data.	0
		0	Send I data to Q data	1 = ignores Q data from interface and disables the clocks to Q datapath. Sends I data to both I and Q DACs.	0
HB1 Control	1C	2:1	HB1[1:0]	00 = input signal not modulated, filter pass band is from −0.4 to +0.4 of f_{IN1}. 01 = input signal not modulated, filter pass band is from 0.1 to 0.9 of f_{IN1}. 10 = input signal modulated by f_{IN1}, filter pass band is from 0.6 to 1.4 of f_{IN1}. 11 = input signal modulated by f_{IN1}, filter pass band is from 1.1 to 1.9 of f_{IN1}.	0
		0	Bypass HB1	1 = bypasses first stage interpolation filter.	0
HB2 Control	1D	6:1	HB2[5:0]	Modulation mode for I Side Half-Band Filter 2. 000000 = input signal not modulated, filter pass band is from −0.25 to +0.25 of f_{IN2}. 001001 = input signal not modulated, filter pass band is from 0.0 to 0.5 of f_{IN2}. 010010 = input signal not modulated, filter pass band is from 0.25 to 0.75 of f_{IN2}. 011011 = input signal not modulated, filter pass band is from 0.5 to 1.0 of f_{IN2}. 100100 = input signal modulated by f_{IN2}, filter pass band is from 0.75 to 1.25 of f_{IN2}. 101101 = input signal modulated by f_{IN2}, filter pass band is from 1.0 to 1.5 of f_{IN2}. 110110 = input signal modulated by f_{IN2}, filter pass band is from 1.25 to 1.75 of f_{IN2}. 111111 = input signal modulated by f_{IN2}, filter pass band is from 1.5 to 2.0 of f_{IN2}.	0
		0	Bypass HB2	1 = bypasses second stage interpolation filter.	0

Reg Name	Addr (Hex)	Bit	Name	Description	Default
HB3 Control	1E	6:1	HB3[5:0]	Modulation mode for I Side Half-Band Filter 3. 000000 = input signal not modulated, filter pass band is from −0.2 to +0.2 of f_{IN3}. 001001 = input signal not modulated, filter pass band is from 0.05 to 0.45 of f_{IN3}. 010010 = input signal not modulated, filter pass band is from 0.3 to 0.7 of f_{IN3}. 011011 = input signal not modulated, filter pass band is from 0.55 to 0.95 of f_{IN3}. 100100 = input signal modulated by f_{IN3}, filter pass band is from 0.8 to 1.2 of f_{IN3}. 101101 = input signal modulated by f_{IN3}, filter pass band is from 1.05 to 1.45 of f_{IN3}. 110110 = input signal modulated by f_{IN3}, filter pass band is from 1.3 to 1.7 of f_{IN3}. 111111 = input signal modulated by f_{IN3}, filter pass band is from 1.55 to 1.95 of f_{IN3}.	0
		0	Bypass HB3	1 = bypasses third stage interpolation filter.	0
Chip ID	1F	7:0	Chip ID[7:0]	This register identifies the device as an AD9122.	8
FTW LSB	30	7:0	FTW[7:0]	See Register 0x33.	0
FTW	31	7:0	FTW[15:8]	See Register 0x33.	0
FTW	32	7:0	FTW[23:16]	See Register 0x33.	0
FTW MSB	33	7:0	FTW[31:24]	FTW[31:0] is the 32-bit frequency tuning word that determines the frequency of the complex carrier generated by the on-chip NCO. The frequency is not updated when the FTW registers are written. The values are only updated when Bit 0 of Register 0x36 transitions from 0 to 1.	0
NCO Phase Offset LSB	34	7:0	NCO Phase Offset[7:0]	See Register 0x35.	0
NCO Phase Offset MSB	35	7:0	NCO Phase Offset[15:8]	The NCO sets the phase of the complex carrier signal when the NCO is reset. The phase offset spans between 0° and 360°. Each bit represents an offset of 0.0055°. Value is in twos complement format.	0
NCO FTW Update	36	5	FRAME FTW acknowledge	1 = indicates that the NCO has been reset due to an extended FRAME pulse signal.	0
		4	FRAME FTW request	0 → 1 = the NCO is reset on the first extended FRAME pulse after this bit transitions from 0 to 1.	0
		1	Update FTW acknowledge	1 = indicates that the FTW has been updated.	0
		0	Update FTW request	0 → 1 = the FTW is updated on 0-to-1 transition of this bit.	0
I Phase Adj LSB	38	7:0	I Phase Adj[7:0]	See Register 0x39.	0
I Phase Adj MSB	39	1:0	I Phase Adj[9:8]	I Phase Adj[9:0] is used to insert a phase offset between the I and Q datapaths. This can be used to correct for phase imbalance in a quadrature modulator. See the Quadrature Phase Correction section for details.	0
Q Phase Adj LSB	3A	7:0	Q Phase Adj[7:0]	See Register 0x3B.	0
Q Phase Adj MSB	3B	1:0	Q Phase Adj[9:8]	Q Phase Adj[9:0] is used to insert a phase offset between the I and Q datapaths. This can be used to correct for phase imbalance in a quadrature modulator. See the Quadrature Phase Correction section for details.	0

Reg Name	Addr (Hex)	Bit	Name	Description	Default
I DAC Offset LSB	3C	7:0	I DAC Offset[7:0]	I DAC Offset[15:0] is a value added directly to the samples written to the I DAC.	0
I DAC Offset MSB	3D	7:0	I DAC Offset[15:8]	See Register 0x3C.	0
Q DAC Offset LSB	3E	7:0	Q DAC Offset[7:0]	Q DAC Offset[15:0] is a value added directly to the samples written to the Q DAC.	0
Q DAC Offset MSB	3F	7:0	Q DAC Offset[15:8]	See Register 0x3E.	0
I DAC FS Adjust	40	7:0	I DAC FS Adj[7:0]	I DAC FS Adj[9:0] sets the full-scale current of the I DAC. The full-scale current can be adjusted from 8.64 mA to 31.6 mA in step sizes of approximately 22.5 µA. 0x000 = 8.64 mA. … 0x200 = 20.14 mA. … 0x3FF = 31.66 mA.	F9
I DAC Control	41	7	I DAC sleep	1 = puts the I-channel DAC into sleep mode (fast wake-up mode).	0
		1:0	I DAC FS Adj[9:8]	See Register 0x40.	1
Aux DAC I Data	42	7:0	I Aux DAC[7:0]	I Aux DAC[9:0] sets the magnitude of the auxiliary DAC current. The range is 0 mA to 2 mA and the step size is 2 µA. 0x000 = 0.000 mA. 0x001 = 0.002 mA. … 0x3FF = 2.046 mA.	0
I Aux DAC Control	43	7	I aux DAC sign	0 = the auxiliary DAC I sign is positive, and the current is directed to the IOUT1P pin (Pin 67). 1 = the auxiliary DAC I sign is negative, and the current is directed to the IOUT1N pin (Pin 66).	0
		6	I aux DAC current direction	0 = the auxiliary DAC I sources current. 1 = the auxiliary DAC I sinks current.	0
		5	I aux DAC sleep	I channel auxiliary DAC sleep.	0
		1:0	I Aux DAC[9:8]	See Register 0x42.	0
Q DAC FS Adj.	44	7:0	Q DAC FS Adj[7:0]	Q DAC FS Adj[9:0] sets the full-scale current of the I DAC. The full-scale current can be adjusted from 8.64 mA to 31.6 mA in step sizes of approximately 22.5 µA. 0x000 = 8.64 mA. … 0x200 = 20.14 mA. … 0x3FF = 31.66 mA.	F9
Q DAC Control	45	7	Q DAC sleep	1 = puts the Q-channel DAC into sleep mode (fast wake-up mode).	0
		1:0	Q DAC FS Adj[9:8]	See Register 0x44.	1

Reg Name	Addr (Hex)	Bit	Name	Description	Default
Aux DAC Q Data	46	7:0	Q Aux DAC[7:0]	Q Aux DAC[9:0] sets the magnitude of the aux DAC current. The range is 0 mA to 2 mA and the step size is 2 µA. 0x000 = 0.000 mA. 0x001 = 0.002 mA. … 0x3FF = 2.046 mA.	0
Q Aux DAC Control	47	7	Q aux DAC sign	0 = the auxiliary DAC Q sign is positive, and the current is directed to the IOUT2P pin (Pin 58). 1 = the auxiliary DAC Q sign is negative, and the current is directed to the IOUT2N pin (Pin 59).	0
		6	Q aux DAC current direction	0 = the auxiliary DAC Q sources current. 1 = the auxiliary DAC Q sinks current.	0
		5	Q aux DAC sleep	Q-channel auxiliary DAC sleep	0
		1:0	Q Aux DAC[9:8]	See Register 0x46.	0
Die Temp Range Control	0x48	6:4	FS Current[2:0]	Auxiliary ADC full-scale current. 000 = lowest current. … 111 = highest current.	0
		3:1	Reference Current[2:0]	Auxiliary ADC reference current. 000 = lowest current. 111 = highest current.	1
		0	Capacitor value	Auxiliary ADC internal capacitor value. 0 = 5 pF. 1 = 10 pF.	0
Die Temp LSB	49	7:0	Die Temp[7:0]	See Register 0x4A.	R
Die Temp MSB	4A	7:0	Die Temp[15:8]	Die Temp[15:0] indicates the approximate die temperature. 0xADCC = −39.9°C 0xC422 = 25.1°C … 0xD8A8 = 84.8°C (see Temperature Sensor section for details)	R
SED Control	67	7	SED compare enable	1 = enables the SED circuitry. None of the flags in this register or the values in Register 0x70 through Register 0x73 are significant if the SED is not enabled.	0
		5	Sample error detected	1 = indicates an error is detected. The bit remains set until cleared. Any write to this register clears this bit to 0.	0
		3	Autoclear enable	1 = enables autoclear mode. This activates Bit 1 and Bit 0 of this register and causes Register 0x70 through Register 0x73 to be autocleared whenever eight consecutive sample data sets are received error free.	0
		1	Compare fail	1 = indicates an error has been detected. This bit remains high until it is autocleared by the reception of eight consecutive error free comparisons or is cleared by writing to this register.	0
		0	Compare pass	1 = indicates that the last sample comparison was error free.	0
Compare I0 LSBs	68	7:0	Compare Value I0[7:0]	Compare Value I0[15:0] is the word that is compared with the I0 input sample captured at the input interface.	B6
Compare I0 MSBs	69	7:0	Compare Value I0[15:8]	See Register 0x68.	7A
Compare Q0 LSBs	6A	7:0	Compare Value Q0[7:0]	Compare Value Q0[15:0] is the word that is compared with the Q0 input sample captured at the input interface.	45

Reg Name	Addr (Hex)	Bit	Name	Description	Default
Compare Q0 MSBs	6B	7:0	Compare Value Q0[15:8]	See Register 0x6A	EA
Compare I1 LSBs	6C	7:0	Compare Value I1[7:0]	Compare Value I1[15:0] is the word that is compared with the I1 input sample captured at the input interface.	16
Compare I1 MSBs	6D	7:0	Compare Value I1[15:8]	See Register 0x6C.	1A
Compare Q1 LSBs	6E	7:0	Compare Value Q1[7:0]	Compare Value Q1[15:0] is the word that is compared with the Q1 input sample captured at the input interface.	C6
Compare Q1 MSBs	6F	7:0	Compare Value Q1[15:8]	See Register 0x6E.	AA
SED I LSBs	70	7:0	Errors Detected I_BITS[7:0]	Errors Detected I_BITS[15:0] indicates which bits were received in error.	0
SED I MSBs	71	7:0	Errors Detected I_BITS[15:8]	See Register 0x70.	0
SED Q LSBs	72	7:0	Errors Detected Q_BITS[7:0]	Errors Detected Q_BITS[15:0] indicates which bits were received in error.	0
SED Q MSBs	73	7:0	Errors Detected Q_BITS[15:8]	See Register 0x72.	0
Revision	7F	5:2	Revision[3:0]	This value corresponds to the die revision number. 0001: Die Revision 1 0010: Die Revision 2	N/A

LVDS INPUT DATA PORTS

The AD9122 has one LVDS data port that receives data for both the I and Q transmit paths. The device can accept data in word, byte, and nibble formats. In word, byte, and nibble modes, the data is sent over 16-bit, 8-bit, and 4-bit LVDS data busses, respectively. The pin assignment of the bus in each mode is shown in Table 12.

Table 12. Data Bit Pair Assignments for Data Input Modes

Mode	MSB, …, LSB
Word	D15, D14, …, D0
Byte[1]	D14, D12, D10, D8, D7, D5, D3, D1
Nibble[1]	D10, D8, D7, D5

[1] In byte and nibble modes, the unused pins can be left floating.

The data is accompanied by a reference bit (DCI) that is used to generate a double data rate (DDR) clock. In byte and nibble modes, a FRAME signal is required for controlling to which DAC the data is sent. All of the interface signals are time aligned. While there is a maximum skew requirement on the bus, there are no setup and hold times to be met.

WORD INTERFACE MODE

In word mode, the DCI signal is a reference bit used for generating the data sampling clock. Time align the DCI signal with the data. The I DAC data should correspond with DCI being high and the Q DAC data with DCI being low, as illustrated in Figure 43.

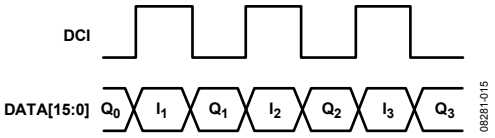

Figure 43. Timing Diagram for Word Mode

BYTE INTERFACE MODE

In byte mode, the DCI signal is a reference bit used for generating the data sampling clock and should be time aligned with the data. The most significant byte of the data should correspond with DCI being high, and the least significant byte of the data should correspond with DCI being low. The FRAME signal indicates to which DAC the data is sent. When FRAME is high, data is sent to the I DAC, and when FRAME is low, data is sent to the Q DAC. The complete timing diagram is shown in Figure 44.

NIBBLE INTERFACE MODE

In nibble mode, the DCI signal is a reference bit used for generating the data sampling clock and should be time aligned with the data. The FRAME signal indicates to which DAC the data is sent. When FRAME is high, data is sent to the I DAC. When FRAME is low, data is sent to the Q DAC. All four nibbles must be written to the device for proper operation. For 12-bit resolution devices, the data in the fourth nibble acts as a place holder for the data framing structure. The complete timing diagram is shown in Figure 45.

FIFO OPERATION

The AD9122 contains a 2-channel, 16-bit wide, eight-word deep FIFO designed to relax the timing relationship between the data arriving at the DAC input ports and the internal DAC data rate clock. The FIFO acts as a buffer that absorbs timing variations between the data source and DAC, such as the clock-to-data variation of an FPGA or ASIC, which significantly increases the timing budget of the interface.

Figure 46 shows the block diagram of the datapath through the FIFO. The data is latched into the device, is formatted, and is then written into the FIFO register determined by the FIFO write pointer. The value of the write pointer is incremented every time a new word is loaded into the FIFO. Meanwhile, data is read from the FIFO register determined by the read pointer and fed into the digital datapath. The value of the read pointer is updated every time data is read into the datapath from the FIFO. The FIFO pointers are incremented at the data rate (DACCLK rate divided by the interpolation ratio).

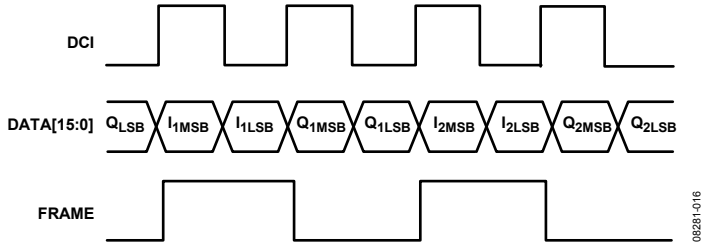

Figure 44. Timing Diagram for Byte Mode

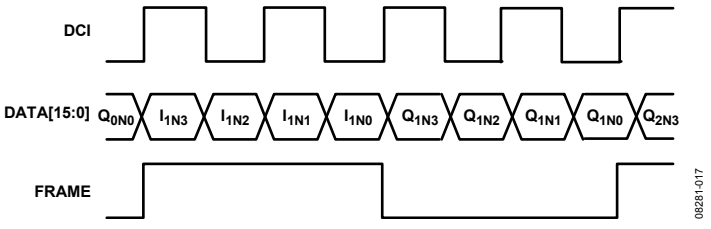

Figure 45. Timing Diagram for Nibble Mode

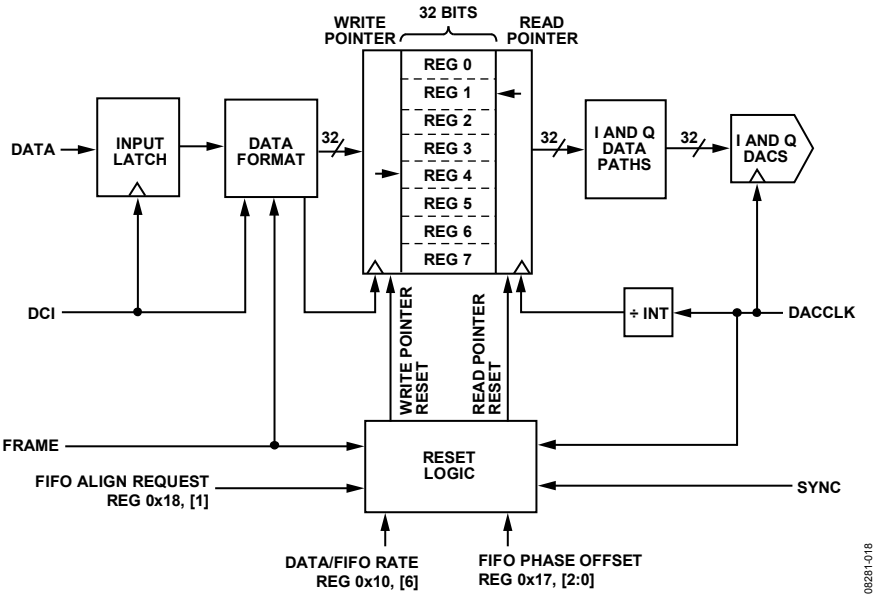

Figure 46. Block Diagram of FIFO

Valid data is transmitted through the FIFO as long as the FIFO does not overflow or become empty. An overflow or empty condition of the FIFO occurs when the write pointer and read pointers point to the same FIFO location. This simultaneous access of data leads to unreliable data transfer through the FIFO and must be avoided.

Nominally, data is written to and read from the FIFO at the same rate. This keeps the FIFO depth constant. If data is written to the FIFO faster than data is read out, the FIFO depth increases. If the data is written to the device slower than data is read, the FIFO depth decreases. For optimum timing margin, the FIFO depth should be maintained near half full (a difference of four between the write pointer and read pointer values). The FIFO depth represents the FIFO pipeline delay and is part of the overall latency of the AD9122.

Resetting the FIFO

To avoid a concurrent read and write to the same FIFO address and assure a fixed pipeline delay, it is important to reset the FIFO pointers to known states. The FIFO pointers can be initialized in two ways: via a write sequence to the serial port or by strobing the FRAME input. There are two types of FIFO resets, a relative reset and an absolute reset. A relative reset enforces a defined FIFO depth. An absolute reset enforces a particular write pointer value when the reset is initiated. A serial port initiated FIFO reset is always a relative reset. A FRAME strobe initiated reset can be either a relative or an absolute reset.

The operation of the FRAME initiated FIFO reset depends on the synchronization mode chosen. When synchronization is disabled, or when configured for data rate mode synchronization, the FRAME strobe initiates a relative FIFO reset. When FIFO mode synchronization is chosen, the FRAME strobe initiates an absolute FIFO reset. More details on the synchronization function can be found in the Multichip Synchronization section.

A summary of the synchronization modes and the type of FIFO reset employed is listed in Table 13.

Table 13. Summary of FIFO Resets

	Synchronization Mode		
FIFO Reset Signal	**Disabled**	**Data Rate**	**FIFO Rate**
Serial Port	Relative	Relative	Relative
FRAME	Relative	Relative	Absolute

Serial Port Initiated FIFO Reset

A serial port initiated FIFO reset can be issued in any mode and always results in a relative FIFO reset. To initialize the FIFO data level through the serial port, Bit 1 of Register 0x18 should be toggled from 0 to 1 and back. When the write to the register is complete, the FIFO data level is initialized. When the initialization is triggered, the next time the read pointer becomes 0, the write pointer is set to the value of the FIFO start level (Register 0x17, Bits[2:0]) variable upon initialization. By default, this is 4 but can be programmed to a value of 0 to 7.

The recommended procedure for a serial port FIFO data level initialization is as follows:

- Request FIFO level reset by setting Register 0x18, Bit 1 to 1.
- Verify that the part acknowledges the request by ensuring Register 0x18, Bit 2 is 1.
- Remove the request by setting Register 0x18, Bit 1 to 0.
- Verify the part drops the acknowledge signal by ensuring Register 0x18, Bit 2 is 0.

FRAME Initiated Relative FIFO Reset

The primary function of the FRAME input is to indicate to which DAC the input data is written. Another function of the FRAME input is initializing the FIFO data level value. This is done by asserting the FRAME signal high for at least the time interval needed to load complete data to the I and Q DACs.

This corresponds to one DCI period in word mode, two DCI periods in byte mode, and four DCI periods in nibble mode.

To initiate a relative FIFO reset with the FRAME signal, the device must be configured in data rate mode (Register 0x10[6]). When FRAME is asserted in data rate mode, the write pointer is set to 4 (by default or to the FIFO start level) the next time the read pointer becomes 0 (see Figure 47).

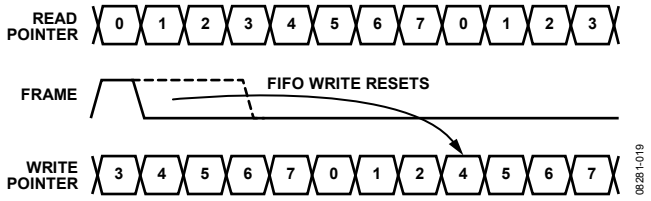

Figure 47. FRAME Input vs. Write Pointer Value, Data Rate Mode

FRAME Initiated Absolute FIFO Reset

In FIFO rate synchronization mode, the REFCLK/SYNC signal is used to reset the FIFO read pointer to Address 0. The edge of the DAC clock used to sample the SYNC signal is selected by Bit 3 of Register 0x10. The FRAME signal is used to reset the FIFO write pointer. In the FIFO rate synchronization mode, the FIFO write pointer is reset immediately after the FRAME signal is asserted high for at least the time interval needed to load complete data to the I and Q DACs. The FIFO write-pointer is reset to the value of the FIFO Phase Offset[2:0], in Register 0x17. FIFO rate synchronization is selected by setting Bit 6 of Register 0x10 to 0.

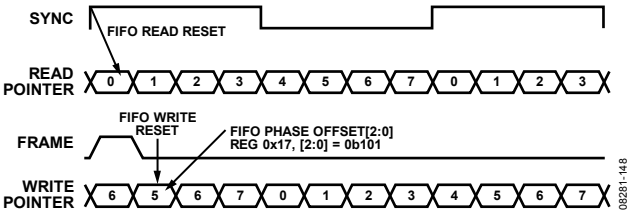

Figure 48. FRAME Input vs. Write Pointer Value, FIFO Rate Mode

Monitoring the FIFO Status

The FIFO initialization and status can be read from Register 0x18. This register provides information on the FIFO initialization method, and whether the initialization was successful. The MSB of Register 0x18 is a FIFO warning flag that can optionally trigger a device IRQ. This flag is an indication that the FIFO is close to emptying (FIFO level is 1) or overflowing (FIFO level is 7). This is an indication that data may soon be corrupted, and action should be taken.

The FIFO data level can be read from Register 0x19 at any time. The serial port reported FIFO data level is denoted as a 7-bit thermometer code of the write counter state relative to the absolute read counter being at 0. The optimum FIFO data level of 4 is therefore reported as a value of 00001111 in the status register. It should be noted that, depending on the timing relationship between DCI and the main DACCLK, the FIFO level value can be off by ±1 count. Therefore, it is important to keep the difference between the read and write pointers to at least two.

INTERFACE TIMING

The timing diagram for the digital interface port is shown in Figure 49. The sampling point of the data bus nominally occurs 350 ps after each edge of the DCI signal and has an uncertainty of ±300 ps, as illustrated by the sampling interval shown in Figure 49. The DATA and FRAME signals must be valid throughout this sampling interval. The DATA and FRAME signals may change at any time between sampling intervals.

The setup (t_S) and hold (t_H) times, with respect to the edges, are shown in Figure 49. The minimum setup and hold times are shown in Table 14.

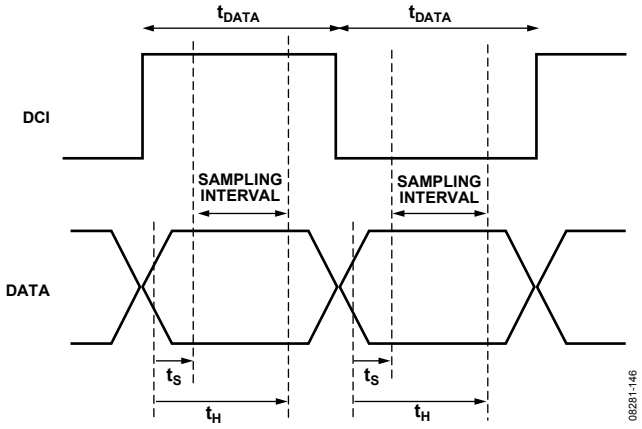

Figure 49. Timing Diagram for Input Data Ports

Table 14. DATA to DCI Setup and Hold Times vs. DCI Delay Value

DCI_DELAY Register 0x16, Bits[1:0]	Minimum Setup Time (t_S) ns	Minimum Hold Time (t_H) ns	Sampling Interval ns
00	−0.05	0.65	0.6
01	−0.23	0.95	0.72
10	−0.38	1.22	0.84
11	−0.47	1.38	0.91

The data interface timing can be verified by using the sample error detection (SED) circuitry. See the Interface Timing Validation section for details.

In data rate mode, a second timing constraint between DCI and DACCLK must be met in addition to the DCI-to-DATA timing shown in Table 15. In data rate mode, only one FIFO slot is being used. The DCI to DACCLK timing restriction is required to prevent data being written to and read from the FIFO slot at the same time. The required timing between DCI and DACCLK is shown in Figure 50.

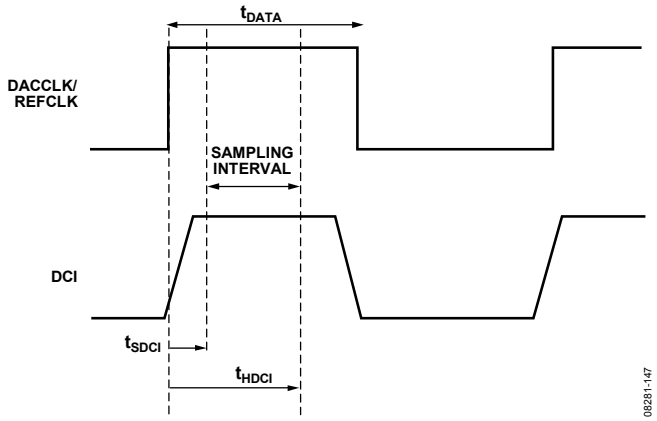

Figure 50. Timing Diagram for Input Data Port (Data Rate Mode)

Table 15. DCI to DACCLK Setup and Hold Times vs. DCI Delay Value

DCI_DELAY Register 0x16, Bits[1:0]	Minimum Setup Time (t_{SDCI}) ns	Minimum Hold Time (t_{HDCI}) ns	Sampling Interval ns
00	−0.07	0.82	0.75
01	−0.24	1.13	0.89
10	−0.39	1.40	1.01
11	−0.49	1.55	1.06

DIGITAL DATAPATH

The block diagram in Figure 51 shows the functionality of the digital datapath. The digital processing includes a premodulation block, three half-band interpolation filters, a quadrature modulator with a fine resolution NCO, phase and offset adjustment blocks, and an inverse sinc filter.

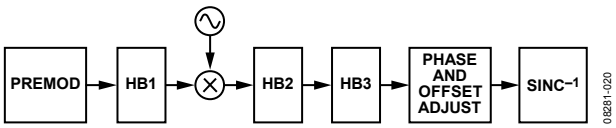

Figure 51. Block Diagram of Digital Datapath

The digital datapath accepts I and Q data streams and processes them as a quadrature data stream. The signal processing blocks can be used when the input data stream is represented as complex data.

The datapath can be used to process an input data stream representing two independent real data streams as well, but the functionality is somewhat restricted. The premodulation block can be used. As well as, any of the nonshifted interpolation filter modes. See the Premodulation section for more details.

PREMODULATION

The half-band interpolation filters have selectable pass bands that allow the center frequencies to be moved in increments of ½ of their input data rate. The premodulation block provides a digital upconversion of the incoming waveform by ½ of the incoming data rate, f_{DATA}. This can be used to frequency shift baseband input data to the center of the interpolation filters pass band.

INTERPOLATION FILTERS

The transmit path contains three interpolation filters. Each of the three interpolation filters provides a 2× increase in output data rate. The half-band (HB) filters can be individually bypassed or cascaded to provide 1×, 2×, 4×, or 8× interpolation ratios. Each of the half-band filter stages offers a different combination of bandwidths and operating modes.

The bandwidth of the three half-band filters with respect to the data rate at the filter input is as follows:

- Bandwidth of HB1 = 0.8 × f_{IN1}
- Bandwidth of HB2 = 0.5 × f_{IN2}
- Bandwidth of HB3 = 0.4 × f_{IN3}

The usable bandwidth is defined as the frequency over which the filters have a pass-band ripple of less than ±0.001 dB and an image rejection of greater than +85 dB. As is discussed in the Half-Band Filter 1 (HB1) section, the image rejection usually sets the usable bandwidth of the filter, not the pass-band flatness.

The half-band filters operate in several modes, providing programmable pass-band center frequencies as well as signal modulation. The HB1 filter has four modes of operation and the HB2 and HB3 filters each have eight modes of operation.

Half-Band Filter 1 (HB1)

HB1 has four modes of operation, as shown in Figure 52. The shape of the filter response is identical in each of the four modes. The four modes are distinguished by two factors, the filter center frequency and whether or not the input signal is modulated by the filter.

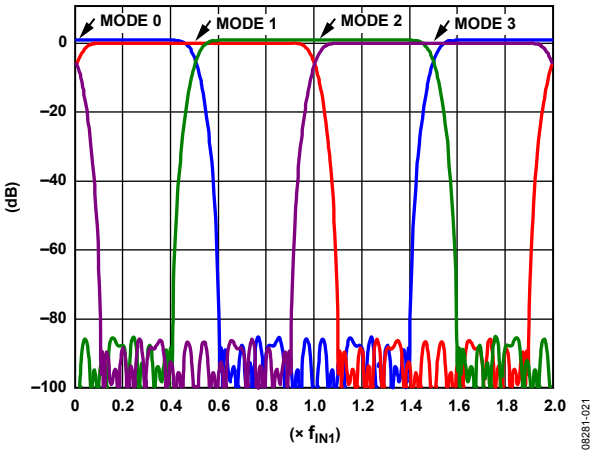

Figure 52. HB1 Filter Modes

As is shown in Figure 52, the center frequency in each mode is offset by ½ of the input data rate (f_{IN1}) of the filter. Mode 0 and Mode 1 do not modulate the input signal. Mode 2 and Mode 3 modulate the input signal by f_{IN1}. When operating in Mode 0 and Mode 2, the I and Q paths operate independently and no mixing of the data between channels occurs. When operating in Mode 1 and Mode 3, mixing of the data between the I and Q paths occurs; therefore, the data input into the filter is assumed complex. Table 16 summarizes the HB1 modes.

Table 16. HB1 Filter Mode Summary

Mode	f_{CENTER}	f_{MOD}	Input Data
0	DC	None	Real or complex
1	$f_{IN}/2$	None	Complex
2	f_{IN}	f_{IN}	Real or complex
3	$3f_{IN}/2$	f_{IN}	Complex

Figure 53 shows the pass-band filter response for HB1. In most applications, the usable bandwidth of the filter is limited by the image suppression provided by the stop-band rejection and not by the pass-band flatness. Table 17 shows the pass-band flatness and stop-band rejection the HB1 filter supports at different bandwidths.

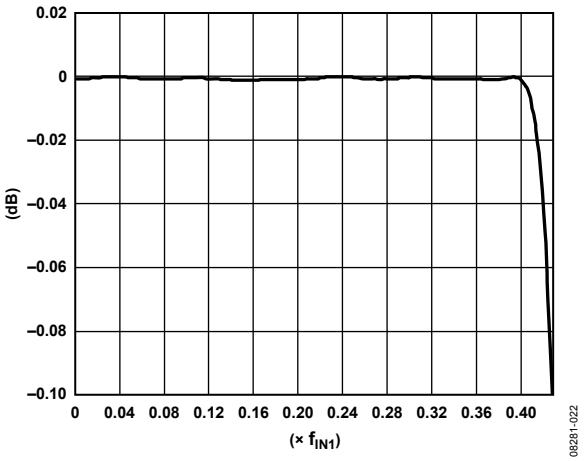

Figure 53. Pass-Band Detail of HB1

Table 17. HB1 Pass-Band and Stop-Band Performance by Bandwidth

Bandwidth (% of f_{IN1})	Pass-Band Flatness (dB)	Stop-Band Rejection (dB)
80	0.001	85
80.4	0.0012	80
81.2	0.0033	70
82.0	0.0076	60
83.6	0.0271	50
85.6	0.1096	40

Half-Band Filter 2 (HB2)

HB2 has eight modes of operation, as shown in Figure 54 and Figure 55. The shape of the filter response is identical in each of the eight modes. The eight modes are distinguished by two factors, the filter center frequency and whether the input signal is modulated by the filter.

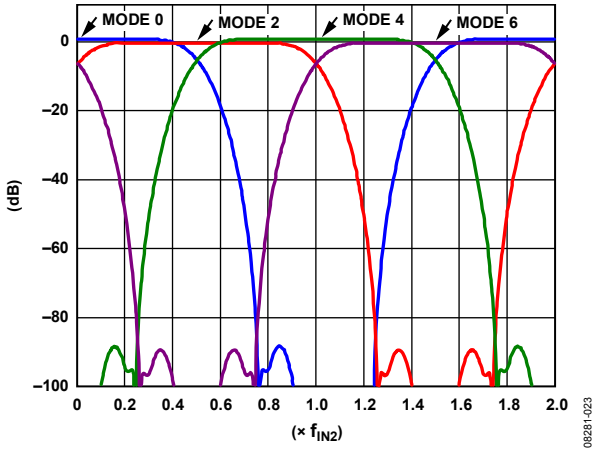

Figure 54. HB2, Even Filter Modes

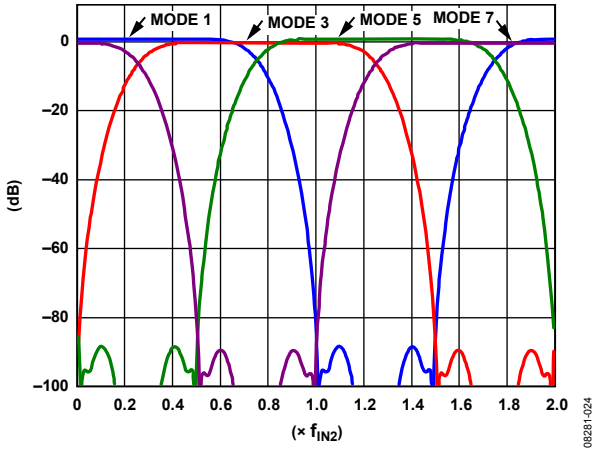

Figure 55. HB2, Odd Filter Modes

As shown in Figure 54 and Figure 55, the center frequency in each mode is offset by ¼ of the input data rate (f_{IN2}) of the filter. Mode 0 through Mode 3 do not modulate the input signal. Mode 4 through Mode 7 modulate the input signal by f_{IN2}. When operating in Mode 0 and Mode 4, the I and Q paths operate independently and no mixing of the data between channels occurs. When operating in the other six modes, mixing of the data between the I and Q paths occurs; therefore, the data input to the filter is assumed complex.

Table 18 summarizes the HB2 and HB3 modes.

Table 18. HB2 and HB3 Filter Mode Summary

Mode	f_{CENTER}	f_{MOD}	Input Data
0	DC	None	Real or complex
1	$f_{IN}/4$	None	Complex
2	$f_{IN}/2$	None	Complex
3	$3f_{IN}/4$	None	Complex
4	f_{IN}	f_{IN}	Real or complex
5	$5f_{IN}/4$	f_{IN}	Complex
6	$3f_{IN}/2$	f_{IN}	Complex
7	$7f_{IN}/4$	f_{IN}	Complex

Figure 56 shows the pass-band filter response for HB2. In most applications, the usable bandwidth of the filter is limited by the image suppression provided by the stop-band rejection and not by the pass-band flatness. Table 19 shows the pass-band flatness and stop-band rejection the HB2 filter supports at different bandwidths.

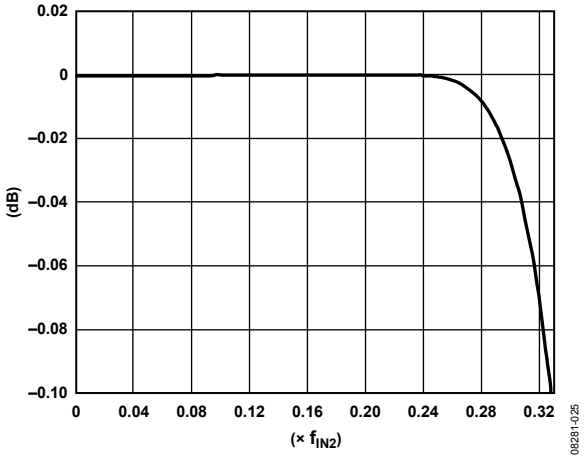

Figure 56. Pass-Band Detail of HB2

Table 19. HB2 Pass-Band and Stop-Band Performance by Bandwidth

Bandwidth (% of f_{IN2})	Pass-Band Flatness (dB)	Stop-Band Rejection (dB)
50	0.001	85
50.8	0.0012	80
52.8	0.0028	70
56.0	0.0089	60
60	0.0287	50
64.8	0.1877	40

Half-Band Filter 3 (HB3)

HB3 has eight modes of operation that function the same as HB2. The primary difference between HB2 and HB3 are the filter bandwidths.

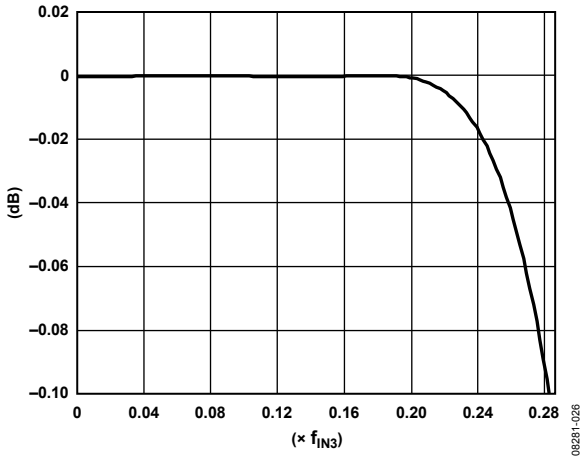

Figure 57. Pass-Band Detail of HB3

Figure 57 shows the pass-band filter response for HB3. In most applications, the usable bandwidth of the filter is limited by the image suppression provided by the stop-band rejection and not by the pass-band flatness. Table 20 shows the pass-band flatness and stop-band rejection the HB3 filter supports at different bandwidths.

Table 20. HB3 Pass-Band and Stop-Band Performance by Bandwidth

Bandwidth (% of f_{IN3})	Pass-Band Flatness (dB)	Stop-Band Rejection (dB)
40	0.001	85
40.8	0.0014	80
42.4	0.002	70
45.6	0.0093	60
49.8	0.03	50
55.6	0.1	40

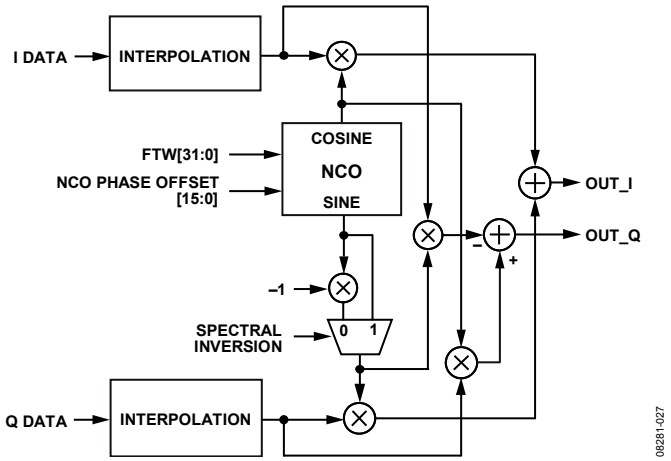

Figure 58. Digital Quadrature Modulator Block Diagram

NCO MODULATION

The digital quadrature modulator makes use of a numerically controlled oscillator, a phase shifter, and a complex modulator to provide a means for modulating the signal by a programmable carrier signal. A block diagram of the digital modulator is shown in Figure 58. The fine modulation provided by the digital modulator, in conjunction with the coarse modulation of the interpolation filters and premodulation block, allows the signal to be placed anywhere in the output spectrum with very fine frequency resolution.

The quadrature modulator is used to mix the carrier signal generated by the NCO with the I and Q signal. The NCO produces a quadrature carrier signal to translate the input signal to a new center frequency. A complex carrier signal is a pair of sinusoidal waveforms of the same frequency, offset 90° from each other. The frequency of the complex carrier signal is set via FTW[31:0] in Register 0x30 through Register 0x33.

The NCO operating frequency, f_{NCO}, is at either f_{DATA} (HB1 bypassed) or twice f_{DATA} (HB1 enabled). The frequency of the complex carrier signal can be set from dc up to f_{NCO}. The frequency tuning word (FTW) is calculated as

$$FTW = \frac{f_{CARRIER}}{f_{NCO}} \times 2^{32}$$

The generated quadrature carrier signal is mixed with the I and Q data. The quadrature products are then summed into the I and Q data paths, as shown in Figure 58.

Updating the Frequency Tuning Word

The frequency tuning word registers are not updated immediately upon writing as other configuration registers do. After loading the FTW registers with the desired values, Bit 0 of Register 0x36 must transition from 0 to 1 for the new FTW to take effect.

DATAPATH CONFIGURATION

Configuring the AD9122 datapath starts with the application requirements of the input data rate, the interpolation ratio, the output signal bandwidth, and the output signal center frequency. Given these four parameters, the first step in configuring the datapath is to verify that the device supports the bandwidth requirements. The modes of the interpolation filters are then chosen. Finally, any additional frequency offset requirements are determined and applied with premodulation and NCO modulation.

Determining Datapath Signal Bandwidth

The available signal bandwidth of the datapath is dependent on the center frequency of the output signal in relation to the center frequency of the interpolation filters used. Signal center frequencies offset from the center frequencies of the half-band filters lower the available signal bandwidth.

When correctly configured, the available complex signal bandwidth for 2× interpolation is always 80% of the input data rate. The available signal bandwidth for 4× interpolation vs. output frequency varies between 50% and 80% of the input data rate, as shown in Figure 59. Note that in 4× interpolation mode, $f_{DAC} = 4 \times f_{DATA}$; therefore, the data shown in Figure 59 repeats four times from dc to f_{DAC}.

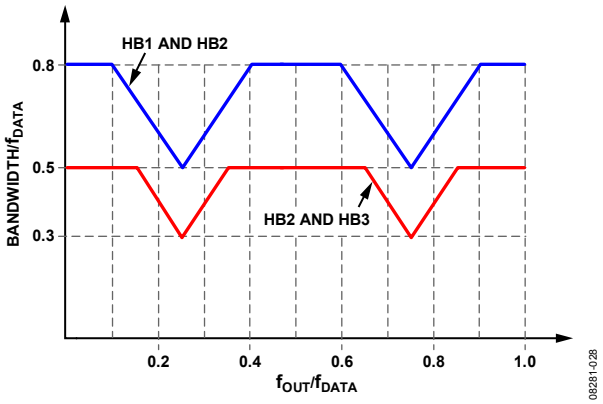

Figure 59. Signal Bandwidth vs. Center Frequency of the Output Signal, 4× Interpolation

Configuring 4× interpolation using the HB2 and HB3 filters can lower the power consumption of the device at the expense of bandwidth. The lower curve in Figure 59 shows that the supported bandwidth in this mode varies from 30% to 50% of f_{DATA}.

The available signal bandwidth for 8× interpolation vs. output frequency varies between 50% and 80% of the input data rate, as shown in Figure 60. Note that in 8× interpolation mode, $f_{DAC} = 8 \times f_{DATA}$; therefore, the data shown in Figure 60 repeats eight times from dc to f_{DAC}.

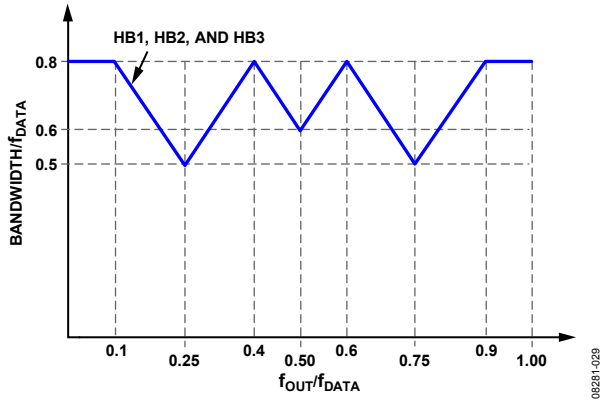

Figure 60. Signal Bandwidth vs. Center Frequency of the Output Signal, 8× Interpolation

DETERMINING INTERPOLATION FILTER MODES

Table 21 shows the recommended interpolation filter settings for a variety of filter interpolation factors, filter center frequencies, and signal modulation. The interpolation modes were chosen based on the final center frequency of the signal and by determining the frequency shift of the signal required. When these are known, and put in terms of the input data rate (f_{DATA}), the filter configuration that comes closest to matching is chosen from Table 21.

Table 21. Recommended Interpolation Filter Modes (Register 0x1C through Register 0x1E)

	Filter Modes				
Interpolation Factor	**HB1[1:0]**	**HB2[5:0]**	**HB3[5:0]**	**f_{SIGNAL} Modulation**	**f_{CENTER} Shift**
8	00 (0)	000000	000000	DC	0
8	01 (1)	001001	000000	DC[1]	$f_{DATA}/2$
8[2]	10 (2)	010010	001001	f_{DATA}	f_{DATA}
8	11 (3)	011011	001001	f_{DATA}[1]	$3f_{DATA}/2$
8	00 (0)	100100	010010	$2f_{DATA}$	$2f_{DATA}$
8	01 (1)	101101	010010	$2f_{DATA}$[1]	$5f_{DATA}/2$
8	10 (2)	110110	011011	$3f_{DATA}$	$3f_{DATA}$
8	11 (3)	111111	011011	3 f_{DATA}[1]	$7f_{DATA}/2$
8	00 (0)	000000	100100	$4f_{DATA}$	$4f_{DATA}$
8	01 (1)	001001	100100	$4f_{DATA}$[1]	$9f_{DATA}/2$
8	10 (2)	010010	101101	$5f_{DATA}$	$5f_{DATA}$
8	11 (3)	011011	101101	$5f_{DATA}$[1]	$11f_{DATA}/2$
8	00 (0)	100100	110110	$6f_{DATA}$	$6f_{DATA}$
8	01 (1)	101101	110110	$6f_{DATA}$[1]	$13f_{DATA}/2$
8	10 (2)	110110	111111	$7f_{DATA}$	$7f_{DATA}$
8	11 (3)	111111	111111	$7f_{DATA}$[1]	$15f_{DATA}/2$
4	00 (0)	000000	Bypass	DC	0
4[3]	01 (1)	001001	Bypass	DC[1]	$f_{DATA}/2$
4	10 (2)	010010	Bypass	f_{DATA}	f_{DATA}
4	11 (3)	011011	Bypass	f_{DATA}[1]	$3f_{DATA}/2$
4	00 (0)	100100	Bypass	$2f_{DATA}$	$2f_{DATA}$
4	01 (1)	101101	Bypass	$2f_{DATA}$[1]	$5f_{DATA}/2$
4	10 (2)	110110	Bypass	$3f_{DATA}$	$3f_{DATA}$
4	11 (3)	111111	Bypass	$3f_{DATA}$[1]	$7f_{DATA}/2$
2	00 (0)	Bypass	Bypass	DC	0
2	01 (1)	Bypass	Bypass	DC[1]	$f_{DATA}/2$
2	10 (2)	Bypass	Bypass	f_{DATA}	f_{DATA}
2	11 (3)	Bypass	Bypass	f_{DATA}[1]	$3f_{DATA}/2$

[1] When HB1 Mode 1 or Mode 3 is used, enabling premodulation provides an addition frequency translation of the input signal by $f_{DATA}/2$, which centers a baseband input signal in the filter pass band.
[2] This configuration was used in the 8× interpolation without NCO example. Also, see the 8× Interpolation Without NCO section.
[3] This configuration was used in the 4× interpolation with NCO example. Also, see the 4× Interpolation With NCO section.

DATAPATH CONFIGURATION EXAMPLE

8× Interpolation Without NCO

Given the following:

f_{DATA} = 100 MSPS

8× interpolation

f_{BW} = 75 MHz

f_{CENTER} = 100 MHz

The desired 75 MHz of bandwidth is 75% of f_{DATA}. In this case, the ratio of f_{OUT}/f_{DATA} = 100/100 = 1.0. From Figure 60, the bandwidth supported at f_{DATA} is 0.8, which verifies that the AD9122 supports the bandwidth required in this configuration.

The signal center frequency is f_{DATA}, and assuming the input signal is at baseband, the frequency shift required is also f_{DATA}. Choosing the third row (highlighted by the superscripted number two) of the IF column from Table 21 selects filter modes that give a center frequency of f_{DATA} and a frequency translation of f_{DATA}. The selected modes for the three half-band filters are: HB1, Mode 2; HB2, Mode 2; and HB3, Mode 1. Figure 61 shows how the signal propagates through the interpolation filters.

Because $2 \times f_{IN1} = f_{IN2}$ and $2 \times f_{IN2} = f_{IN3}$, the signal appears frequency scaled by ½ into each consecutive stage. The output signal band spans 0.15 to 0.35 of f_{IN3} (400 MHz). Therefore, the output frequency supported is 60 MHz to 140 MHz, which covers the 75 MHz bandwidth centered at 100 MHz, as desired.

4× Interpolation With NCO

Given the following:

f_{DATA} = 250 MSPS

4× interpolation

f_{BW} = 140 MHz

f_{CENTER} = 175 MHz

The desired 140 MHz of bandwidth is 56% of f_{DATA}. As shown in Figure 59, the value at $0.7 \times f_{DATA}$ is 0.6. This is calculated as $0.8 - 2(0.7 - 0.6) = 0.6$. This verifies that the AD9122 supports a bandwidth of 60% of f_{DATA}, which exceeds the required 56%.

The signal center frequency is $0.7 \times f_{DATA}$, and assuming the input signal is at baseband, the frequency shift required is also $0.7 \times f_{DATA}$. Choosing the second row in the IF column in the 4× interpolation section in Table 21 selects the filter modes that give a center frequency of $f_{DATA}/2$ and no frequency translation. The selected modes for the three half-band filters are HB1, Mode 1; HB2, Mode 1; and HB3, bypassed.

Because Mode 1 of HB1 was selected, the premodulation block should be enabled. This provides $f_{DATA}/2$ modulation, which centers the baseband input data at the center frequency of HB1. The digital modulator can be used to provide the final frequency translation of $0.2 \times f_{DATA}$ to place the output signal at $0.7 \times f_{DATA}$, as desired.

The formula for calculating the FTW of the NCO is:

$$FTW = \frac{f_{CARRIER}}{f_{NCO}} \times 2^{32}$$

where:
$f_{CARRIER} = 0.2 \times f_{DATA}$.
$f_{NCO} = 2 \times f_{DATA}$. Therefore, FTW = $2^{32}/10$.

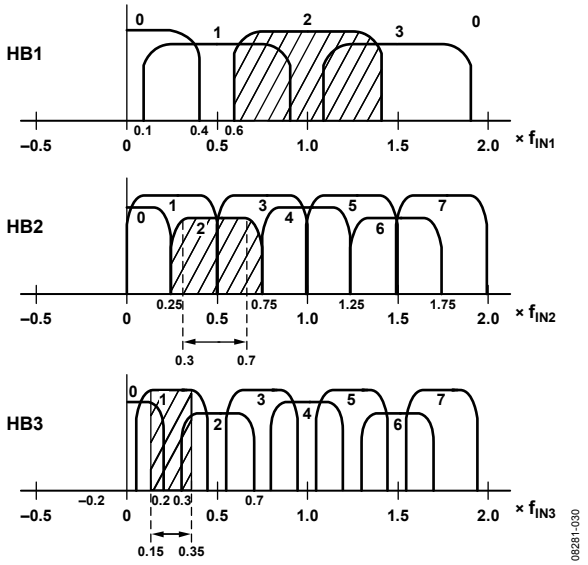

Figure 61. Signal Propagation for 8× Interpolation (f_{DATA} Modulation)

DATA RATES vs. INTERPOLATION MODES

Table 23 summarizes the maximum bus speed (f_{BUS}), supported input data rates, and signal bandwidths with the various combinations of bus width modes and interpolation rates. The real signal bandwidth supported is a fraction of the input data rate, which depends on the interpolation filters (HB1, HB2, or HB3) selected. The complex signal bandwidth supported is twice the real signal bandwidth.

In general, 2× interpolation is best supported by enabling HB1, and 4× interpolation is best supported enabling HB1 and HB2. In some cases, power dissipation can be lowered by avoiding HB1. If the bandwidth required is low enough, 2× interpolation can be supported by using HB2, and 4× interpolation can be supported by using HB2 and HB3.

COARSE MODULATION MIXING SEQUENCES

The coarse digital quadrature modulation occurs within the interpolation filters. The modulation shifts the frequency spectrum of the incoming data by the frequency offset selected. The frequency offsets available are multiples of the input data rate. The modulation is equivalent to multiplying the quadrature input signal by a complex carrier signal, C(t), of the form

$$C(t) = \cos(\omega_c t) + j \sin(\omega_c t)$$

In practice, this modulation results in mixing functions as shown in Table 22.

Table 22. Modulation Mixing Sequences

Modulation	Mixing Sequence
$f_S/2$	I = I, –I, I, –I, … Q = Q, –Q, Q, –Q, …
$f_S/4$	I = I, Q, –I, –Q, … Q = Q, –I, –Q, I, …
3 $f_S/4$	I = I, –Q, –I, Q, … Q = Q, I, –Q, –I, …
$f_S/8$	I = I, r(I + Q), Q, r(–I + Q), –I, –r(I + Q), –Q, r(I – Q), … Q = Q, r(Q – I), –I, –r(Q + I), –Q, r(–Q + I),I, r(Q + I), …

Note that $r = \frac{\sqrt{2}}{2}$

As shown in Table 22, the mixing functions of most of the modes crosscouple samples between the I and Q channels. The I and Q channels only operate independently in $f_S/2$ mode. This means that real modulation using both the I and Q DAC outputs can only be done in $f_S/2$ mode. All other modulation modes require complex input data and produce complex output signals.

Table 23. Summary of Data Rates and Bandwidths vs. Interpolation Modes (DVDD18 = 1.8V+/-2%)

	Filter Modes						
Bus Width	HB3	HB2	HB1	f_{BUS} (Mbps)	f_{DATA} (Mbps)	Real Signal Bandwidth (MHz)	f_{DAC} (MHz)
Nibble (4 Bits)	0	0	0	1200	150	75	150
	0	0	1	1200	150	60	300
	0	1	0	1200	150	37.5	300
	0	1	1	1200	150	60	600
	1	1	0	1200	150	37.5	600
	1	1	1	1200	150	60	1200
Byte (8 Bits)	0	0	0	1200	300	150	300
	0	0	1	1200	300	120	600
	0	1	0	1200	300	75	600
	0	1	1	1200	300	120	1200
	1	1	0	1200	300	75	1200
	1	1	1	600	150	60	1200
Word (16 Bits)	0	0	0	1200	600	300	600
	0	0	1	800	400	160	800
	0	1	0	1200	600	150	1200
	0	1	1	600	300	120	1200
	1	1	0	600	300	75	1200
	1	1	1	300	150	60	1200

QUADRATURE PHASE CORRECTION

The purpose of the quadrature phase correction block is to enable compensation of the phase imbalance of the analog quadrature modulator following the DAC. If the quadrature modulator has a phase imbalance, the unwanted sideband appears with significant energy. Tuning the quadrature phase adjust value can optimize image rejection in single sideband radios.

Ordinarily, the I and Q channels have an angle of precisely 90° between them. The quadrature phase adjustment is used to change the angle between the I and Q channels. When the I Phase Adj[9:0] is set to 1000000000b, the I DAC output moves approximately 1.75° away from the Q DAC output, creating an angle of 91.75° between the channels. When the I Phase Adj[9:0] is set to 0111111111b, the I DAC output moves approximately 1.75° toward the Q DAC output, creating an angle of 88.25° between the channels.

The Q Phase Adj[9:0] works in a similar fashion. When the Q Phase Adj[9:0] is set to 1000000000b, the Q DAC output moves approximately 1.75° away from the I DAC output, creating an angle of 91.75° between the channels. When the Q Phase Adj[9:0] is set to 0111111111b, the Q DAC output moves approximately 1.75° toward the I DAC output, creating an angle of 88.25° between the channels.

Based on these two endpoints, the combined resolution of the phase compensation register is approximately 3.5°/1024 or 0.00342° per code.

DC OFFSET CORRECTION

The dc value of the I datapath and the Q datapath can be independently controlled by adjusting the I DAC Offset[15:0] and Q DAC Offset[15:0] values in Register 0x3C through Register 0x3F. These values are added directly to the datapath values. Care should be taken not to overrange the transmitted values.

Figure 62 shows how the DAC offset current varies as a function of the I DAC Offset[15:0] and the Q DAC Offset[15:0] values. With the digital inputs fixed at midscale (0x0000, twos complement data format), Figure 62 shows the nominal I_{OUTP} and I_{OUTN} currents as the DAC offset value is swept from 0 to 65535. Because I_{OUTP} and I_{OUTN} are complementary current outputs, the sum of I_{OUTP} and I_{OUTN} is always 20 mA.

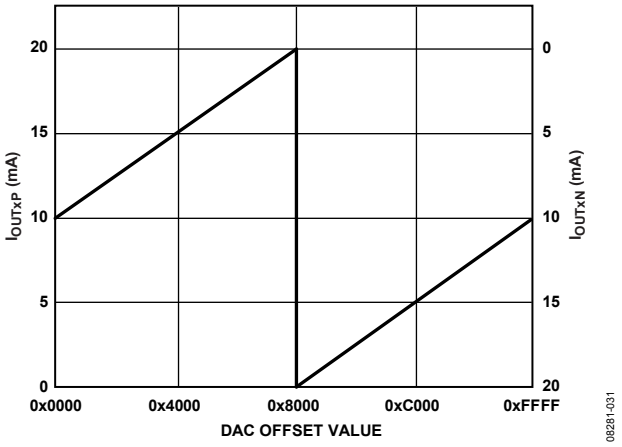

Figure 62. DAC Output Currents vs. DAC Offset Value

INVERSE SINC FILTER

The inverse sinc ($sinc^{-1}$) filter is a nine-tap FIR filter. The composite response of the $sinc^{-1}$ and the sin(x)/x response of the DAC is shown in Figure 62. The composite response has less than ±0.05 dB pass-band ripple up to a frequency of $0.4 \times f_{DACCLK}$. To provide the necessary peaking at the upper end of the pass band, the inverse sinc filters shown have an intrinsic insertion loss of about 3.2 dB. Figure 63 shows the composite frequency response.

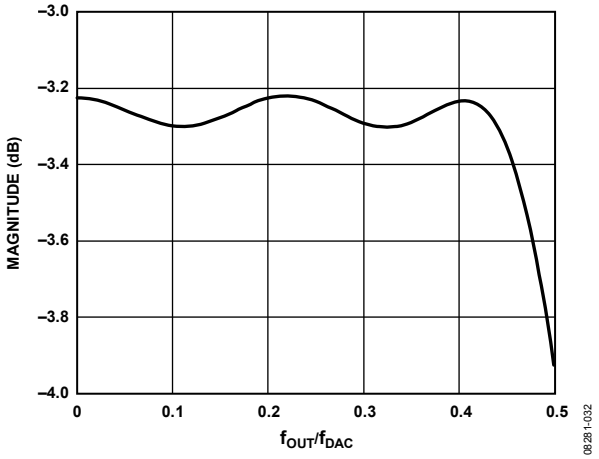

Figure 63. Sample Composite Responses of the $Sinc^{-1}$ Filter with Sin(x)/x Roll-Off

The $sinc^{-1}$ filter is enabled by default. It can be bypassed by setting the bypass $sinc^{-1}$ bit (Register 0x1B, Bit 6).

DAC INPUT CLOCK CONFIGURATIONS

DAC INPUT CLOCK CONFIGURATIONS

The AD9122 DAC sample clock (DACCLK) can be sourced directly or by clock multiplying. Clock multiplying employs the on-chip phased-locked loop (PLL) that accepts a reference clock operating at a submultiple of the desired DACCLK rate, most commonly the data input frequency. The PLL then multiplies the reference clock up to the desired DACCLK frequency, which can then be used to generate all the internal clocks required by the DAC. The clock multiplier provides a high quality clock that meets the performance requirements of most applications. Using the on-chip clock multiplier removes the burden of generating and distributing the high speed DACCLK.

The second mode bypasses the clock multiplier circuitry and allows DACCLK to be sourced directly to the DAC core. This mode enables the user to source a very high quality clock directly to the DAC core. Sourcing the DACCLK directly through the REFCLKP, REFCLKN, DACCLKP, and DACCLKN pins may be necessary in demanding applications that require the lowest possible DAC output noise, particularly when directly synthesizing signals above 150 MHz.

Driving the DACCLK and REFCLK Inputs

The REFCLK and DACCLK differential inputs share similar clock receiver input circuitry. Figure 64 shows a simplified circuit diagram of the input. The on-chip clock receiver has a differential input impedance of about 10 kΩ. It is self biased to a common-mode voltage of about 1.25 V. The inputs can be driven by direct coupling differential PECL or LVDS drivers. The inputs can also be ac-coupled if the driving source cannot meet the input compliance voltage of the receiver.

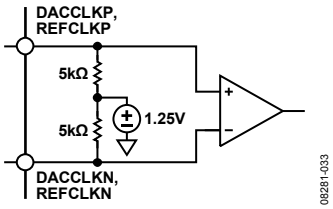

Figure 64. Clock Receiver Input Equivalent Circuit

The minimum input drive level to either of the clock inputs is 200 mV p-p differential. The optimal performance is achieved when the clock input signal is between 800 mV p-p differential and 1.6 V p-p differential. Whether using the on-chip clock multiplier or sourcing the DACCLK, directly, it is necessary that the input clock signal to the device has low jitter and fast edge rates to optimize the DAC noise performance.

Direct Clocking

Direct clocking with a low noise clock produces the lowest noise spectral density at the DAC outputs. To select the differential CLK inputs as the source for the DAC sampling clock, set the PLL enable bit (Register 0x0A, Bit[7]) to 0. This powers down the internal PLL clock multiplier and selects the input from the DACCLKP and DACCLKN pins as the source for the internal DAC sample clock.

The device also has duty-cycle correction circuitry and differential input level correction circuitry. Enabling these circuits can provide improved performance in some cases. The control bits for these functions can be found in Register 0x08. See Table 11 for complete details.

Clock Multiplication

The on-chip PLL clock multiplier circuit can be used to generate the DAC sample rate clock from a lower frequency reference clock. When the PLL enable bit (Register 0x0A, Bit[7]) is set to 1, the clock multiplication circuit generates the DAC sample clock from the lower rate REFCLK input. The functional diagram of the clock multiplier is shown in Figure 65.

The clock multiplication circuit operates such that the VCO outputs a frequency, f_{VCO}, equal to the REFCLK input signal frequency multiplied by N1 × N0.

$$f_{VCO} = f_{REFCLK} \times (N1 \times N0)$$

The DAC sample clock frequency, f_{DACCLK}, is equal to

$$f_{DACCLK} = f_{REFCLK} \times N1$$

The output frequency of the VCO must be chosen to keep f_{VCO} in the optimal operating range of 1.0 GHz to 2.1 GHz. The frequency of the reference clock and the values of N1 and N0 must be chosen so that the desired DACCLK frequency can be synthesized and the VCO output frequency is in the correct range.

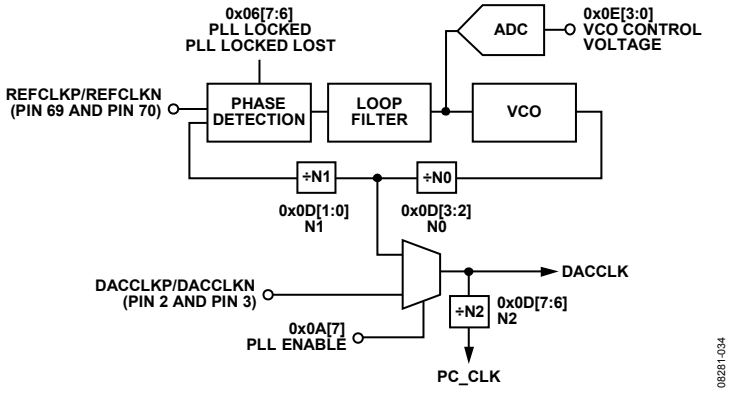

Figure 65. PLL Clock Multiplication Circuit

PLL Settings

There are three settings for the PLL circuitry that should be programmed to their nominal values. The PLL values shown in Table 24 are the recommended settings for these parameters.

Table 24. PLL Settings

PLL SPI Control	Address Register	Bit	Optimal Setting
PLL Loop Bandwidth[1:0]	0x0C	[7:6]	11
PLL Charge Pump Current[4:0]	0x0C	[4:0]	10001
PLL Cross Control Enable	0x0D	[4]	1

Configuring the VCO Tuning Band

The PLL VCO has a valid operating range from approximately 1.0 GHz to 2.1 GHz covered in 63 overlapping frequency bands. For any desired VCO output frequency, there may be several valid PLL band select values. The frequency bands of a typical device are shown in Figure 66. Device-to-device variations and operating temperature affect the actual band frequency range. Therefore, it is required that the optimal PLL band select value be determined for each individual device.

Automatic VCO Band Select

The device has an automatic VCO band select feature on chip. Using the automatic VCO band select feature is a simple and reliable method of configuring the VCO frequency band. This feature is enabled by starting the PLL in manual mode, then placing the PLL in auto band select mode. This is done by setting Register 0x0A to a value of 0xCF, then to a value of 0xA0. When these values are written, the device executes an automated routine that determines the optimal VCO band setting for the device. The setting selected by the device ensures that the PLL remains locked over the full −40°C to +85°C operating temperature range of the device without further adjustment. (The PLL remains locked over the full temperature range even if the temperature during initialization is at one of the temperature extremes.)

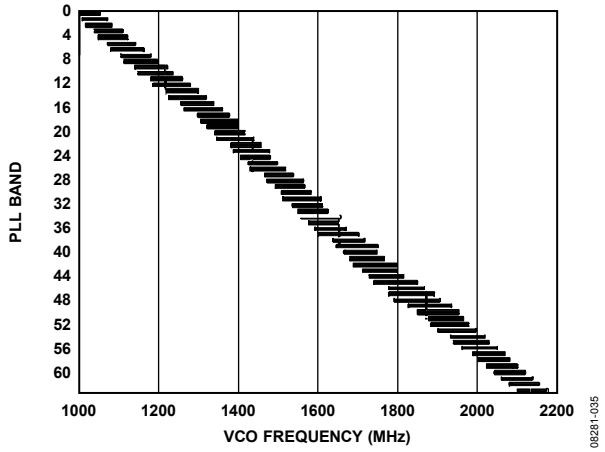

Figure 66. PLL Lock Range Over Temperature for a Typical Device

Manual VCO Band Select

The device also has a manual band select mode (PLL manual enable, Register 0x0A, Bit[6] = 1) that allows the user to select the VCO tuning band. When in manual mode, the VCO band is set directly with the value written to the manual VCO band, (Register 0x0A, Bit[5:0]). To properly select the VCO band, follow these steps:

1. Put the device in manual band select mode.
2. Sweep the VCO band over a range of bands that result in the PLL being locked.
3. For each band, verify that the PLL is locked and read the PLL using the VCO control voltage (Register 0x0E[3:0]).
4. Select the band that results in the control voltage being closest to the center of the range (that is, 0000 or 1000). See Table 25 for more details. The resulting VCO band should be the optimal setting for the device. Write this band to the manual VCO band (Register 0x0A[5:0]) value.
5. If desired, an indication of where the VCO is within the operating frequency band can be determined by querying the VCO control voltage. Table 25 shows how to interpret the PLL VCO control voltage (Register 0x0E, Bits[2:0]) value.

Table 25. VCO Control Voltage Range Indications

VCO Control Voltage	Indication
1111 1110	Move to higher VCO band
1101 1100 1011 1010	VCO is operating in the higher end of frequency band
1001 1000 0111 0110	VCO is operating within an optimal region of the frequency band
0101 0100 0011 0010	VCO is operating in the lower end of frequency band
0001 0000	Move to lower VCO band

ANALOG OUTPUTS

TRANSMIT DAC OPERATION

Figure 67 shows a simplified block diagram of the transmit path DACs. The DAC core consists of a current source array, a switch core, digital control logic, and full-scale output current control. The DAC full-scale output current (I_{OUTFS}) is nominally 20 mA. The output currents from the IOUT1P/IOUT2P and IOUT1N/IOUT2N pins are complementary, meaning that the sum of the two currents always equals the full-scale current of the DAC. The digital input code to the DAC determines the effective differential current delivered to the load.

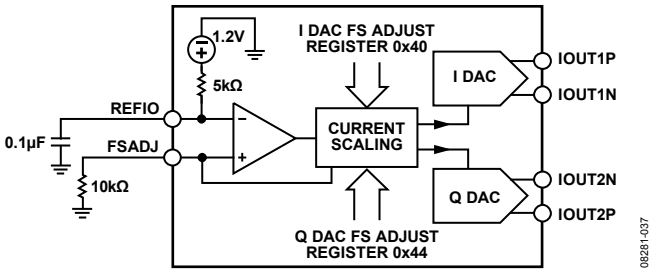

Figure 67. Simplified Block Diagram of DAC Core

The DAC has a 1.2 V band gap reference with an output impedance of 5 kΩ. The reference output voltage appears on the REFIO pin. When using the internal reference, decouple the REFIO pin to AVSS with a 0.1 µF capacitor. Only use the internal reference for external circuits that draw dc currents of 2 µA or less. For dynamic loads or static loads greater than 2 µA, buffer the REFIO pin. If desired, an external reference (between 1.10 V and 1.30 V) can be applied to the REFIO pin. The internal reference can either be overdriven or powered down by setting Register 0x43, Bit [5].

A 10 kΩ external resistor, R_{SET}, must be connected from the FSADJ pin to AVSS. This resistor, along with the reference control amplifier, sets up the correct internal bias currents for the DAC. Because the full-scale current is inversely proportional to this resistor, the tolerance of R_{SET} is reflected in the full-scale output amplitude.

The full-scale current equation, where the DAC gain is set individually for the I and Q DACs in Register 040 and Register 044, respectively, follows:

$$I_{FS} = \frac{V_{REF}}{R_{SET}} \times \left(72 + \left(\frac{3}{16} \times DAC\ gain\right)\right)$$

For nominal values of V_{REF} (1.2 V), R_{SET} (10 kΩ), and DAC gain (512), the full-scale current of the DAC is typically 20.16 mA. The DAC full-scale current can be adjusted from 8.66 mA to 31.66 mA by setting the DAC gain parameter setting as shown in Figure 68.

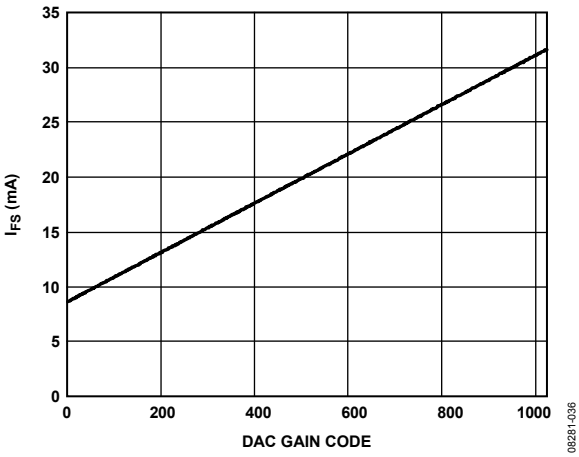

Figure 68. DAC Full-Scale Current vs. DAC Gain Code

Transmit DAC Transfer Function

The output currents from the IOUT1P/IOUT2P and IOUT1N/IOUT2N pins are complementary, meaning that the sum of the two currents always equals the full-scale current of the DAC. The digital input code to the DAC determines the effective differential current delivered to the load. IOUT1P/IOUT2P provide maximum output current when all bits are high. The output currents vs. DACCODE for the DAC outputs are expressed as

$$I_{OUTP} = \left[\frac{DACCODE}{2^N}\right] \times I_{OUTFS} \quad (1)$$

$$I_{OUTN} = I_{OUTFS} - I_{OUTP} \quad (2)$$

where $DACCODE = 0$ to $2^N - 1$.

Transmit DAC Output Configurations

The optimum noise and distortion performance of the AD9122 is realized when it is configured for differential operation. The common-mode error sources of the DAC outputs are significantly reduced by the common-mode rejection of a transformer or differential amplifier. These common-mode error sources include even-order distortion products and noise. The enhancement in distortion performance becomes more significant as the frequency content of the reconstructed waveform increases and/or its amplitude increases. This is due to the first-order cancellation of various dynamic common-mode distortion mechanisms, digital feedthrough, and noise.

Figure 69 shows the most basic DAC output circuitry. A pair of resistors, R_O, is used to convert each of the complementary output currents to a differential voltage output, V_{OUT}. Because the current outputs of the DAC are high impedance, the differential driving point impedance of the DAC outputs, R_{OUT}, is equal to $2 \times R_O$. Figure 70 illustrates the output voltage waveforms.

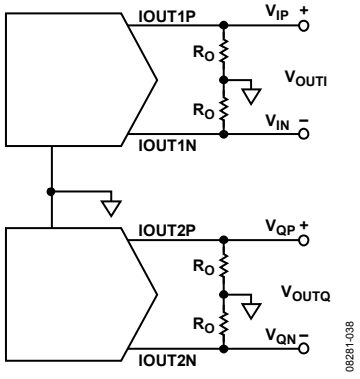

Figure 69. Basic Transmit DAC Output Circuit

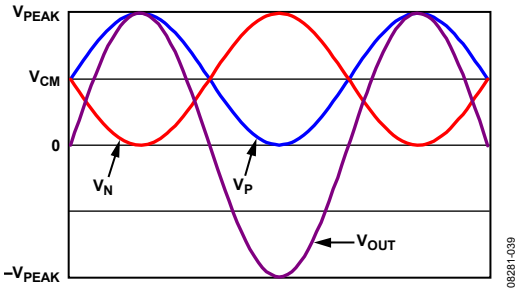

Figure 70. Voltage Output Waveforms

The common-mode signal voltage, V_{CM}, is calculated as

$$V_{CM} = \frac{I_{FS}}{2} \times R_O$$

The peak output voltage, V_{PEAK}, is calculated as

$$V_{PEAK} = I_{FS} \times R_O$$

With this circuit configuration, the single-ended peak voltage is the same as the peak differential output voltage.

Transmit DAC Linear Output Signal Swing

To achieve optimum performance, the DAC outputs have a linear output compliance voltage range that must be adhered to. The linear output signal swing is dependent on the full-scale output current, I_{OUTFS}, and the common-mode level of the output. Figure 71 and Figure 72 show the IMD performance vs. the common-mode voltage at the different full-scale currents and output frequencies.

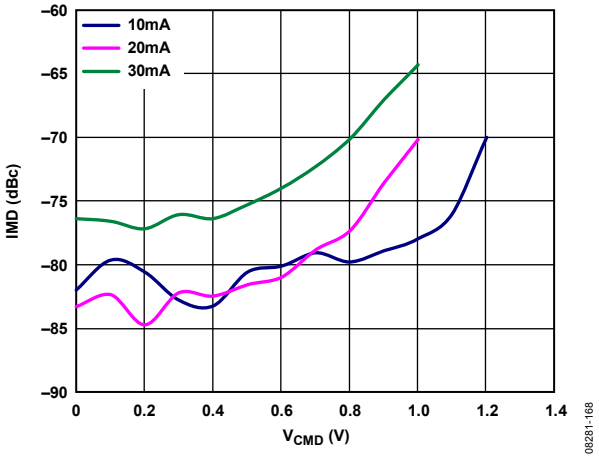

Figure 71. IMD vs. Output Common-Mode Voltage (f_{OUT} = 61 MHz, R_{LOAD} = 50 Ω differential, I_{FS} = 10 mA, 20 mA, and 30 mA)

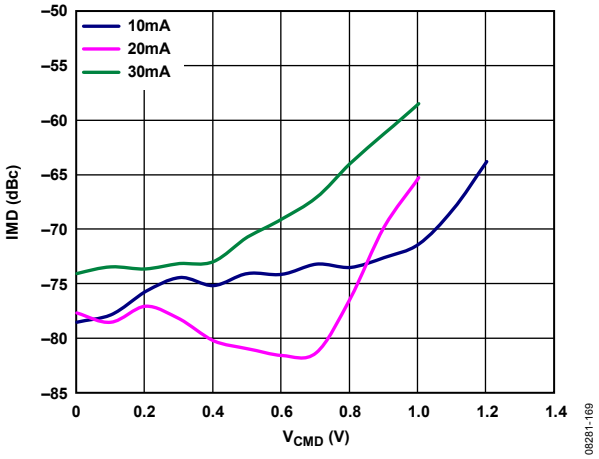

Figure 72. IMD vs. Output Common-Mode Voltage (f_{OUT} = 161 MHz, R_{LOAD} = 50 Ω differential, I_{FS} = 10 mA, 20 mA, and 30 mA)

AUXILIARY DAC OPERATION

The AD9122 have two auxiliary DACs, one associated with the I path and one associated with the Q path. These auxiliary DACs can be used to compensate for dc offsets in the transmitted signal. Each auxiliary DAC has a single-ended current that can sink or source current into either the P or N output of the associated transmit DAC. The auxiliary DAC structure is shown in Figure 73.

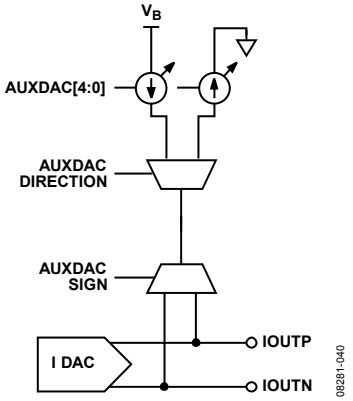

Figure 73. Auxiliary DAC Structure

The control registers for controlling the I and Q auxiliary DACs are in Register 0x42, Register 0x43, and Register 0x46.

Interfacing to Modulators

The AD9122 interfaces to the ADL537x family of modulators with a minimal number of components. An example of the recommended interface circuitry is shown in Figure 74.

The baseband inputs of the ADL537x family require a dc bias of 500 mV. The nominal midscale output current on each output of the DAC is 10 mA (½ the full-scale current). Therefore, a single 50 Ω resistor to ground from each of the DAC outputs results in the desired 500 mV dc common-mode bias for the inputs to the ADL537x. The signal level can be reduced through the addition of the load resistor in parallel with the modulator inputs. The peak-to-peak voltage swing of the transmitted signal is

$$V_{SIGNAL} = I_{FS} \times \frac{[2 \times R_B \times R_L]}{[2 \times R_B + R_L]}$$

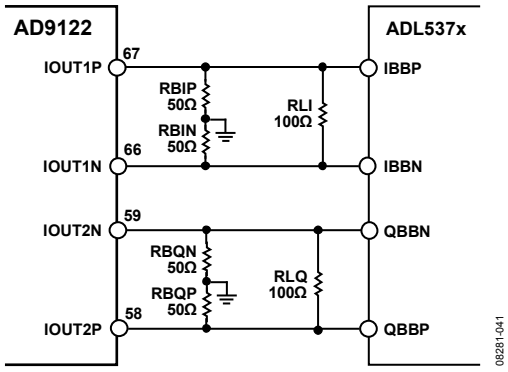

Figure 74. Typical Interface Circuitry Between the AD9122 and the ADL537x Family of Modulators

BASEBAND FILTER IMPLEMENTATION

Most applications require a baseband anti-imaging filter between the DAC and the modulator to filter out Nyquist images and broadband DAC noise. The filter can be inserted between the I-V resistors at the DAC output and the signal-level setting resistor across the modulator input. Doing this establishes the input and output impedances for the filter.

Figure 76 shows a fifth-order, low-pass filter. A common-mode choke is used between the I-V resistors and the remainder of the filter. This removes the common-mode signal produced by the DAC and prevents the common-mode signal from being converted to a differential signal, which can appear as unwanted spurious signals in the output spectrum. Splitting the first filter capacitor into two and grounding the center point creates a common-mode low-pass filter, providing additional common-mode rejection of high frequency signals. A purely differential filter can pass common-mode signals.

DRIVING THE ADL5375-15

The ADL5375-15 requires a 1500 mV dc bias and, therefore, requires a slightly more complex interface than most other Analog Devices, Inc, modulators. It is necessary to level shift the DAC output from a 500 mV dc bias to the 1500 mV dc bias that the ADL5375-15 requires. Level shifting can be achieved with a purely passive network, as shown in Figure 75. In this network, the dc bias of the DAC remains at 500 mV while the input to the ADL5375-15 is 1500 mV. This passive, level shifting network introduces approximately 2 dB of loss in the ac signal.

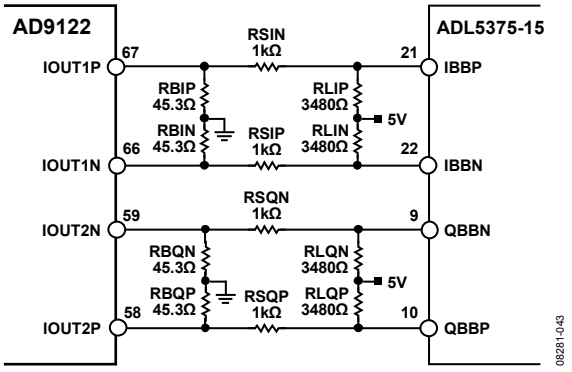

Figure 75. Passive, Level Shifting Network for Biasing ADL5375-15

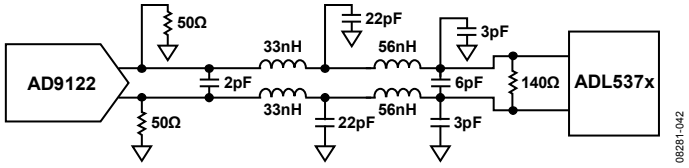

Figure 76. DAC Modulator Interface with Fifth-Order, Low Pass Filter

REDUCING LO LEAKAGE AND UNWANTED SIDEBANDS

Analog quadrature modulators can introduce unwanted signals at the LO frequency due to dc offset voltages in the I and Q baseband inputs, as well as feedthrough paths from the LO input to the output. The LO feedthrough can be nulled by applying the correct dc offset voltages at the DAC output. This can be done using the auxiliary DACs (Register 0x42, Register 0x43, Register 0x46, and Register 0x47) or by using the digital dc offset adjustments (Register 0x3C through Register 0x3F). The advantage of using the auxiliary DACs is that none of the main DAC dynamic range is used to perform the dc offset adjustment.

The disadvantage is that the common-mode level of the output signal changes as a function of the auxiliary DAC current. The opposite is true when the digital offset adjustment is used.

Good sideband suppression requires both gain and phase matching of the I and Q signals. The I/Q phase adjust (Register 0x38 through Register 0x3B) and DAC FS adjust (Register 0x40 and Register 0x44) registers can be used to calibrate I and Q transmit paths to optimize the sideband suppression.

DEVICE POWER DISSIPATION

The AD9122 has four supply rails: AVDD33, IOVDD, DVDD18, and CVDD18.

The AVDD33 supply powers the DAC core circuitry. The power dissipation of the AVDD33 supply rail is independent of the digital operating mode and sample rate. The current drawn from the AVDD33 supply rail is typically 57 mA (188 mW) when the full-scale current of the I and Q DACs is set to the nominal value of 20 mA. Changing the full-scale current directly impacts the supply current drawn from the AVDD33 rail. For example, if the full-scale current of the I DAC and the Q DAC is changed to 10 mA, the AVDD33 supply current drops by 20 mA to 37 mA.

The IOVDD voltage supplies the serial port I/O pins, the $\overline{\text{RESET}}$ pin, and the $\overline{\text{IRQ}}$ pin. The voltage applied to the IOVDD pin can range from 1.8 V to 3.3 V. The current drawn by the IOVDD supply pin is typically 3 mA.

The DVDD18 supply powers all of the digital signal processing blocks of the device. The power consumption from this supply is a function of which digital blocks are enabled and the frequency at which the device is operating.

The CVDD18 supply powers the clock receiver and clock distribution circuitry. The power consumption from this supply varies directly with the operating frequency of the device. CVDD18 also powers the PLL. The power dissipation of the PLL is typically 80 mA when enabled.

Figure 77 through Figure 81 detail the power dissipation of the AD9122 under a variety of operating conditions. All of the graphs are taken with data being supplied to both the I and Q channels. The power consumption of the device does not vary significantly with changes in the coarse modulation mode selected or analog output frequency. Graphs of the total power dissipation are shown along with the power dissipation of the DVDD18 and CVDD18 supplies.

Maximum power dissipation can be estimated to be 20% higher than the typical power dissipation.

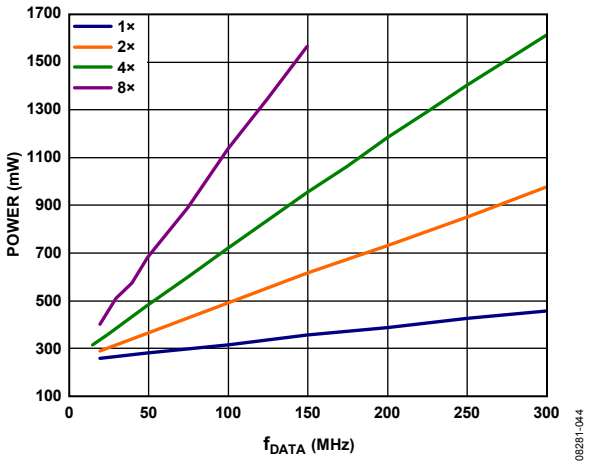

Figure 77. Total Power Dissipation vs. f_{DATA} Without PLL, Fine NCO, and Inverse Sinc

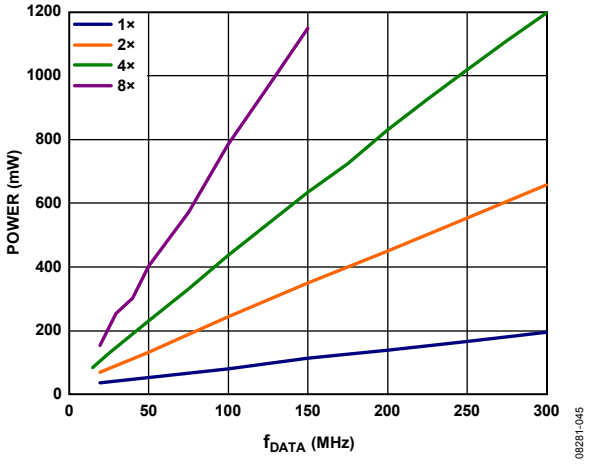

Figure 78. DVDD18 Power Dissipation vs. f_{DATA} Without Fine NCO and Inverse Sinc

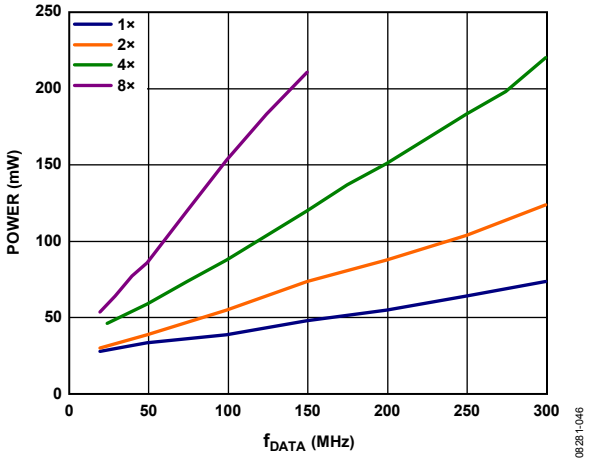

Figure 79. CVDD18 Power Dissipation vs. f_{DATA} with PLL Disabled

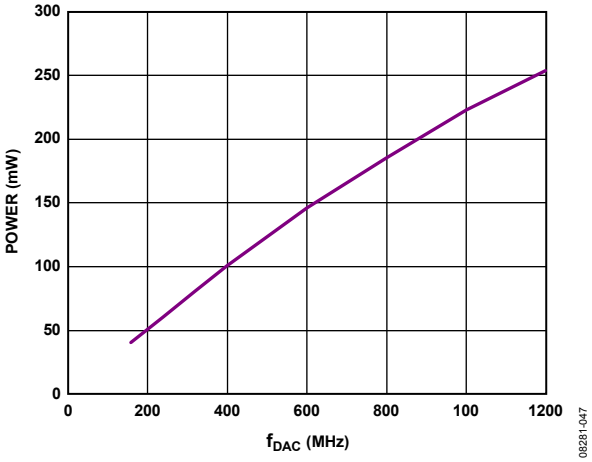

Figure 80. DVDD18 Power Dissipation vs. f_{DAC} Due to Inverse Sinc Filter

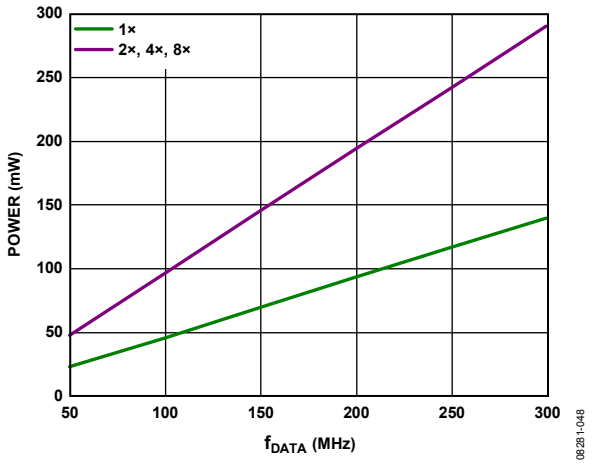

Figure 81. DVDD18 Power Dissipation vs. f_{DATA} Due to Fine NCO

TEMPERATURE SENSOR

The AD9122 has a diode-based temperature sensor for measuring the temperature of the die. The temperature reading is accessed through Register 0x49 and Register 0x4A. The temperature of the die can be calculated by

$$T_{DIE} = \frac{(DieTemp[15:0] - 47{,}925)}{88}$$

where T_{DIE} is the die temperature in °C. The temperature accuracy is ±5°C typical.

Estimates of the ambient temperature can be made if the power dissipation of the device is known. For example, if the device power dissipation is 800 mW, and the measured die temperature is 50°C, then the ambient temperature can be calculated as

$$T_A = T_{DIE} - P_D \times T_{JA} = 50 - 0.8 \times 20.7 = 33.4°C$$

where:
T_A is the ambient temperature in °C.
T_{JA} is the thermal resistance from junction to ambient of the AD9122, as shown in Table 7.

To use the temperature sensor, it must be enabled by setting Register 0x01, Bit 4 to 0. In addition, to obtain accurate readings, the range control register (Register 0x48) should be set to 0x02.

MULTICHIP SYNCHRONIZATION

System demands may require that the outputs of multiple DACs be synchronized with each other or with a system clock. Systems that support transmit diversity or beam forming, where multiple antennas are used to transmit a correlated signal, require multiple DAC outputs to be phase aligned with each other. Systems with a time division multiplexing transmit chain may require one or more DACs to be synchronized with a system-level reference clock.

Multiple devices are considered synchronized to each other when the state of the clock generation state machines is identical for all parts, and when time aligned data is being read from the FIFOs of all parts simultaneously. Devices are considered synchronized to a system clock when there is a fixed and known relationship between the clock generation state machine and the data being read from the FIFO and a particular clock edge of the system clock. The AD9122 has provisions for enabling multiple devices to be synchronized to each other or to a system clock.

The AD9122 supports synchronization in two different modes, data rate mode and FIFO rate mode. The lowest rate clock that the synchronization logic attempts to synchronize to distinguishes these two modes. In data rate mode, the input data rate represents the lowest synchronized clock. In FIFO rate mode, the FIFO rate, which is the data rate divided by the FIFO depth of 8, represents the lowest rate clock. The advantage of FIFO rate synchronization is increased time between keep-out windows for DCI changes relative to the DACCLK or REFCLK input.

When in data rate mode, the elasticity of the FIFO is not used to absorb timing variations between the data source and the DAC, resulting in keep-out widows repeating at the input data rate.

The method chosen for providing the DAC sampling clock directly impacts the synchronization methods available. When the device clock multiplier is used, only data rate mode is available. When the DAC sampling clock is sourced directly, both data rate mode and FIFO rate mode synchronization is available. The following sections detail the synchronization methods for enabling both clocking modes and querying the status of the synchronization logic.

SYNCHRONIZATION WITH CLOCK MULTIPLICATION

When using the clock multiplier to generate the DAC sample rate clock, the REFCLK input signal acts as both the reference clock for the PLL-based clock multiplier and as the synchronization signal. To synchronize devices, distribute the REFCLK signal with low skew to all of the devices that need to be synchronized. Skew between the REFCLK signals of the different devices shows up directly as a timing mismatch at the DAC outputs.

The frequency of the REFCLK signal is typically equal to the input data rate. The FRAME and DCI signals can be created in the FPGA along with the data. A circuit diagram of a typical configuration is shown in Figure 82.

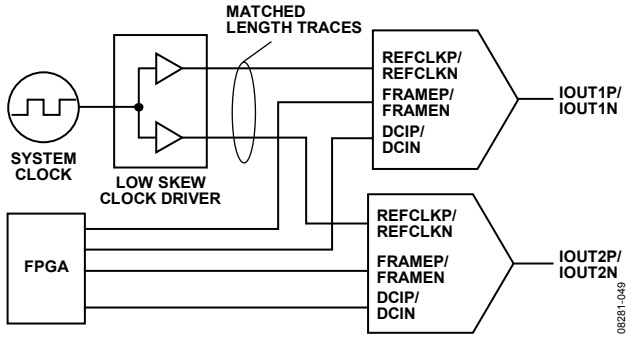

Figure 82. Typical Circuit Diagram for Synchronizing Devices

The Procedure for Synchronization when Using the PLL section outlines the steps required to synchronize multiple devices. The procedure assumes that the REFCLK signal is applied to all of the devices, and that the PLL of each device is phase locked to it. The following procedure must be carried out on each individual device.

Procedure for Synchronization when Using the PLL

Configure for data rate, periodic synchronization by writing 0xC0 to the sync control register (Register 0x10). Additional synchronization options are available.

Read the sync status register (Register 0x12) and verify that the sync locked bit (Bit 6) is set high, indicating that the device achieved back-end synchronization and that the sync lost bit (Bit 7) is low. These levels indicate that the clocks are running with a constant and known phase relative to the sync signal.

Reset the FIFO by strobing the FRAME signal high for the time required to write two complete input data words. Resetting the FIFO ensures that the correct data is being read from the FIFO.

This completes the synchronization procedure, and at this stage, all devices should be synchronized.

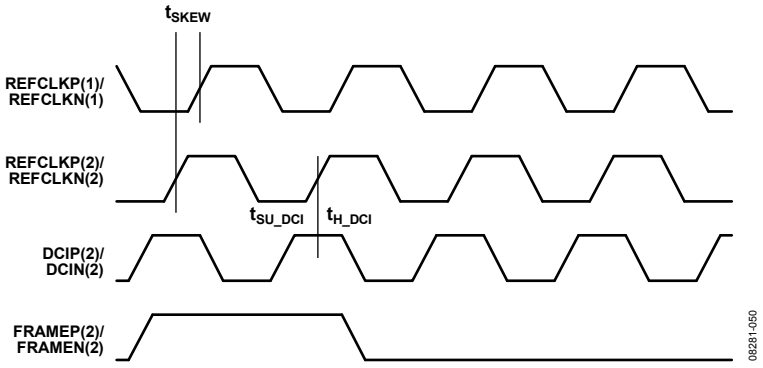

Figure 83. Timing Diagram Required for Synchronizing Devices

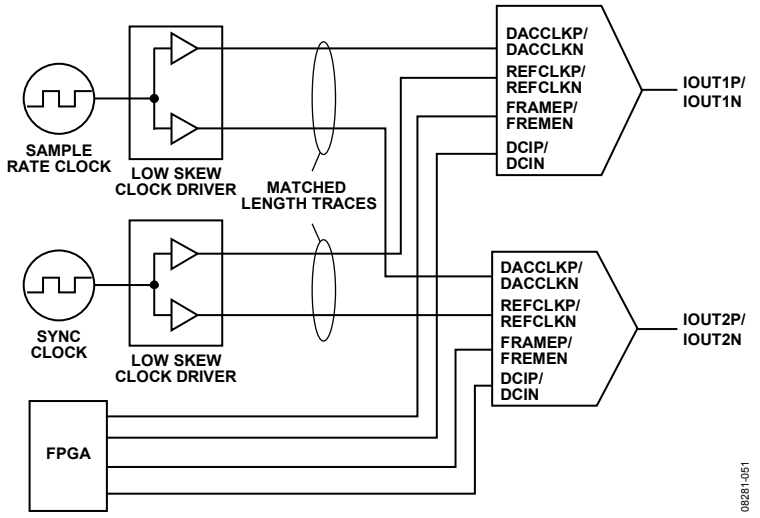

Figure 84. Typical Circuit Diagram for Synchronizing Devices to a System Clock

To maintain synchronization, the skew between the REFCLK signals of the devices must be less than t_{SKEW} nanoseconds. There is also a setup-and-hold time to be observed between the DCI and data of each device and the REFCLK signal. When resetting the FIFO, the FRAME signal must be held high for the time interval required to write two complete input data words. A timing diagram of the input signals is shown in Figure 83.

The preceding example shows a REFCLK frequency equal to the data rate. While this is the most common situation, it is not strictly required for proper synchronization. Any REFCLK frequency that satisfies the following equation is acceptable.

$$f_{SYNC\_I} = f_{DACCLK}/2^N \text{ and } f_{SYNC\_I} \leq f_{DATA}$$

where N = 0, 1, 2, or 3.

As an example, a configuration with 4× interpolation and clock frequencies of f_{VCO} = 1600 MHz, f_{DACCLK} = 800 MHz, f_{DATA} = 200 MHz, and f_{SYNC_I} = 100 MHz is a viable solution.

SYNCHRONIZATION WITH DIRECT CLOCKING

When directly sourcing the DAC sample rate clock, a separate REFCLK input signal is required for synchronization. To synchronize devices, the DACCLK signal and the REFCLK signal must be distributed with low skew to all of the devices being synchronized. If the devices need to be synchronized to a master clock, then use the master clock directly for generating the REFCLK input (see Figure 84).

DATA RATE MODE SYNCHRONIZATION

The Procedure for Data Rate Synchronization when Directly Sourcing the DAC Sampling Clock section outlines the steps required to synchronize multiple devices in data rate mode. The procedure assumes that the DACCLK and REFCLK signals are applied to all of the devices. The procedure must be carried out on each individual device.

Procedure for Data Rate Synchronization when Directly Sourcing the DAC Sampling Clock

Configure for data rate, periodic synchronization by writing 0xC0 to the sync control register (Register 0x10). Additional synchronization options are available and are described in the Additional Synchronization Features section.

Poll the sync locked bit (Register 0x12, Bit 6) to verify that the device is back-end synchronized. A high level on this bit indicates that the clocks are running with a constant and known phase relative to the sync signal.

Reset the FIFO by strobing the FRAME signal high for the time interval required to input two complete data input words. Resetting the FIFO ensures that the correct data is being read from the FIFO of each of the devices simultaneously.

This completes the synchronization procedure, and at this stage, all devices should be synchronized.

To ensure that each of the DACs are updated with the correct data on the same CLK edge, two timing relationships must be met on each DAC. DCIP/DCIN and D[15:0]P/D[15:0]N must meet the setup-and-hold times with respect to the rising edge of DACCLK, and REFCLK must also meet the setup-and-hold time with respect to the rising edge of DACCLK. When resetting the FIFO, the FRAME signal must be held high the time required to input two complete data input words. When these conditions are met, the outputs of the DACs are updated within t_{SKEW} + t_{OUTDLY} nanoseconds of each other. A timing diagram that illustrates the timing requirements of the input signals is shown in Figure 85.

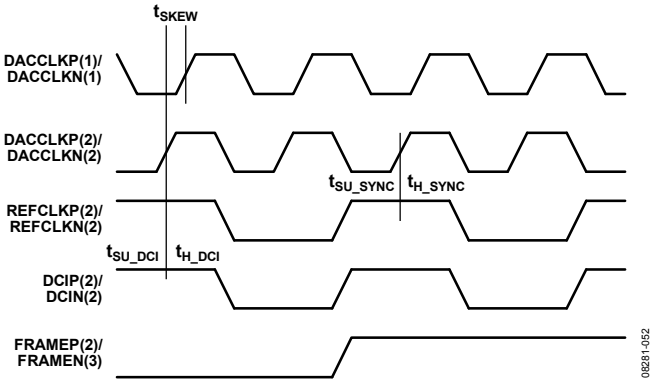

Figure 85. Data Rate Synchronization Signal Timing Requirements, 2x Interpolation

Figure 85 shows the synchronization signal timing with 2× interpolation; therefore, f_{DCI} = ½ × f_{CLK}. The REFCLK input is shown to be equal to the data rate. The maximum frequency at which the device can be resynchronized in data rate mode can be expressed as

$$f_{SYNC\_I} = f_{DATA}/2^N$$

where *N* is any nonnegative integer.

Generally, for values of N equal to or greater than 3, select the FIFO rate synchronization mode.

FIFO RATE MODE SYNCHRONIZATION

The Procedure for FIFO Rate Synchronization when Directly Sourcing the DAC Sampling Clock section outlines the steps required to synchronize multiple devices in FIFO rate mode. The procedure assumes that the REFCLK and DACCLK signals are applied to all of the devices. The procedure must be carried out on each individual device.

Procedure for FIFO Rate Synchronization when Directly Sourcing the DAC Sampling Clock

Configure for FIFO rate, periodic synchronization by writing 0x80 to the sync control register (Register 0x10). Additional synchronization options are available and are described in the Additional Synchronization Features section.

Poll the sync locked bit (Register 0x12, Bit[6]) to verify that the device is back-end synchronized. A high level on this bit indicates that the clocks are running with a constant and known phase relative to the sync signal.

Reset the FIFO by strobing the FRAME signal high for the time required to input to complete input words. Resetting the FIFO ensures that the correct data is being read from the FIFO of each of the devices simultaneously.

This completes the synchronization procedure, and at this stage, all devices should be synchronized.

To ensure that each of the DACs are updated with the correct data on the same CLK edge, two timing relationships must be met on each DAC. DCIP/DCIN and D[15:0]P/D[15:0]N must meet the setup-and-hold times with respect to the rising edge of DACCLK, and REFCLK must also meet the setup-and-hold time with respect to the rising edge of DACCLK. When resetting the FIFO, the FRAME signal must be held high for at least three data periods (that is, 1.5 cycles of DCI). When these conditions are met, the outputs of the DACs are updated within t_{SKEW} + t_{OUTDLY} nanoseconds of each other. A timing diagram that illustrates the timing requirements of the input signals is shown in Figure 86.

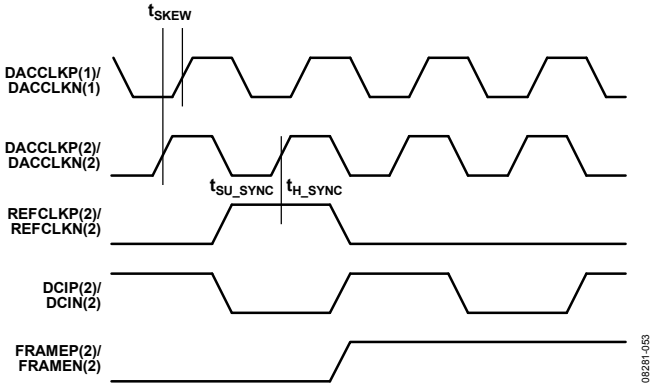

Figure 86. FIFO Rate Synchronization Signal Timing Requirements, 2× Interpolation

Figure 86 shows the synchronization signal timing with 2× interpolation; therefore, f_{DCI} = ½ × f_{CLK}. The REFCLK input is shown to be equal to the FIFO rate. More generally, the maximum frequency at which the device can be resynchronized in FIFO rate mode can be expressed as

$$f_{SYNC\_I} = (f_{DATA}/8 \times 2^N)$$

where *N* is any nonnegative integer.

ADDITIONAL SYNCHRONIZATION FEATURES

The synchronization logic incorporates additional features that provide means for querying the status of the synchronization, improving the robustness of the synchronization, and a one shot synchronization mode. These features are detailed in the Sync Status Bits and Timing Optimization sections that follow.

Sync Status Bits

When the sync locked bit (Register 0x12, Bit 6) is set, it indicates that the synchronization logic has reached alignment. This alignment is determined when the clock generation state machine phase is constant. It takes between (11 + averaging) × 64 and (11 + averaging) × 128 DACCLK cycles. This bit may optionally trigger an $\overline{\text{IRQ}}$ as described in the Interrupt Request Operation section.

When the sync lost bit (Register 0x12, Bit 7) is set, it indicates a previously synchronized device has lost alignment. This bit is latched and remains set until cleared by overwriting the register. This bit may optionally trigger an $\overline{\text{IRQ}}$ as described in the Interrupt Request Operation section.

The sync phase readback bits (Register 0x13, Bits[7:0]) report the current clock phase in a 6.2 format. Bits[7:2] report which of the 64 states (0 to 63) the clock is currently in. When averaging is enabled, Bits[1:0] provide ¼ state accuracy (for 0, ¼, ½, ¾). The lower two bits give an indication of the timing margin issues that may exist. If the sync sampling is error free, the fractional clock state should be 00.

Timing Optimization

The REFCLK signal is sampled by a version of the DACCLK. If sampling errors are being detected, the opposite sampling edge can be selected to improve the sampling point. The sampling edge can be selected by setting Register 0x10, Bit 3 (1 = rising and 0 = falling).

The synchronization logic resynchronizes when a phase change between the REFCLK signal and the state of the clock generation state machine exceeds a threshold. To mitigate the effects of jitter and prevent erroneous resynchronizations, the relative phase can be averaged. The amount of averaging is set by the sync averaging bits (Register 0x10, Bits[2:0]) and can be set from 1 to 128. The higher the number of averages, the more slowly the device recognizes and resynchronizes to a legitimate phase correction. Generally, the averaging should be made as large as possible while still meeting the allotted resynchronization time interval.

The sync phase request bits value (Register 0x11, Bits[5:0]) is the state to which the clock generation state machine resets upon initialization. By varying this value, the timing of the internal clocks, with respect to the REFCLK signal, can be adjusted. Every increment of the Sync Phase Request[5:0] (Register 0x11, Bits[5:0]) value advances the internal clocks by one DACCLK period. This offset can be used for two purposes: to skew the outputs of two synchronized DAC outputs in increments of the DACCLK period and to change the relative timing between the DCI input and REFCLK. This may allow for a more optimal placement of the DCI sampling point in data rate synchronization mode.

Table 26. Synchronization Setup and Hold Times

Parameter	Min	Max	Unit
t_{SKEW}	$-t_{DACCLK}/2$	$+t_{DACCLK}/2$	ps
t_{SV_SYNC}	100		ps
T_{H_SYNC}	330		ps

INTERRUPT REQUEST OPERATION

The AD9122 provides an interrupt request output signal (on Pin 7, $\overline{IRQ}$) that can be used to notify an external host processor of significant device events. Upon assertion of the interrupt, the device should be queried to determine the precise event that occurred. The $\overline{IRQ}$ pin is an open-drain, active low output. Pull the $\overline{IRQ}$ pin high external to the device. This pin can be tied to the interrupt pins of other devices with open-drain outputs to wired-OR these pins together.

Sixteen different event flags provide visibility into the device. These 16 flags are located in the two event flag registers (Register 0x06 and Register 0x07). The behavior of each of the event flags is independently selected in the interrupt enable registers (Register 0x04 and Register 0x05). When the flag interrupt enable is active, the event flag latches and triggers an external interrupt. When the flag interrupt is disabled, the event flag simply monitors the source signal and the external $\overline{IRQ}$ remains inactive.

Figure 87 shows the $\overline{IRQ}$-related circuitry. This diagram shows how the event flag signals propagate to the $\overline{IRQ}$ output. The interupt_enable signal represents one bit from the interrupt enable register. The event_flag_source signal represents one bit from the event flag register. The event_flag_source signal represents one of the device signals that can be monitored such as the PLL_locked signal from the PLL phase detector or the FIFO Warning 1 signal from the FIFO controller.

When an interrupt enable bit is set high, the corresponding event flag bit reflects a positively tripped (that is, latched on the rising edge of the event_flag_source version of the event_flag_source signal. This signal also asserts the external $\overline{IRQ}$. When an interrupt enable bit is set low, the event flag bit reflects the current status of the event_flag_source signal, and the event flag has no effect on the external $\overline{IRQ}$.

The latched version of an event flag (the interupt_source signal) can be cleared in two ways. The recommended way is by writing 1 to the corresponding event flag bit. A hardware or software reset also clears the interupt_source.

INTERRUPT SERVICE ROUTINE

Interrupt request management starts by selecting the set of event flags that require host intervention or monitoring. Those events that require host action should be enabled so that the host is notified when they occur. For events requiring host intervention, upon $\overline{IRQ}$ activation, run the following routine to clear an interrupt request:

1. Read the status of the event flag bits that are being monitored.
2. Set the interupt enable bit low so that the unlatched event_flag_source can be monitored directly.
3. Perform any actions that may be required to clear the event_source_flag. In many cases, no specific actions may be required.
4. Read the event flag to verify the actions taken have cleared the event_flag_source.
5. Clear the interrupt by writing 1 to the event flag bit.
6. Set the interupt enable bits of the events to be monitored.

Note that some of the event_flag_source signals are latched signals. These are cleared by writing to the corresponding event flag bit. Details of each of the event flags can be found in Table 11.

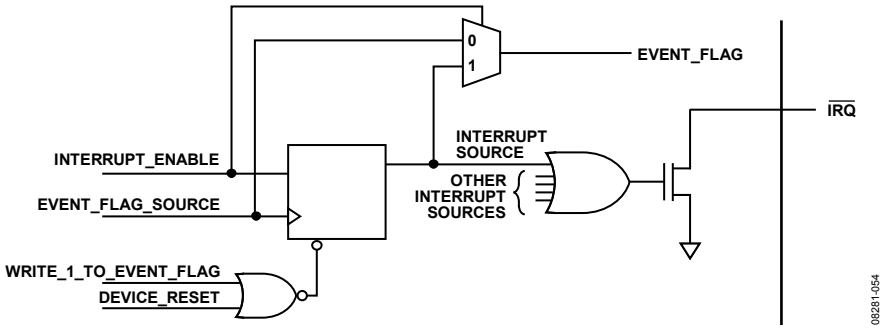

Figure 87. Simplified Schematic of $\overline{IRQ}$ Circuitry

INTERFACE TIMING VALIDATION

The AD9122 provides on-chip sample error detection (SED) circuitry that simplifies verification of the input data interface. The SED compares the input data samples captured at the digital input pins with a set of comparison values. The comparison values are loaded into registers through the SPI port. Differences between the captured values and the comparison values are detected and stored. Options are available for customizing SED test sequencing and error handling.

SED OPERATION

The SED circuitry operates on a data set made up of four 16-bit input words, denoted as I0, Q0, I1, and Q1. To properly align the input samples, the first I data-word (that is, I0) is indicated by asserting FRAME for at least one complete input sample.

Figure 88 shows the input timing of the interface in word mode. The FRAME signal can be issued once at the start of the data transmission, or it can be asserted repeatedly at intervals coinciding with the I0 and Q0 data-words.

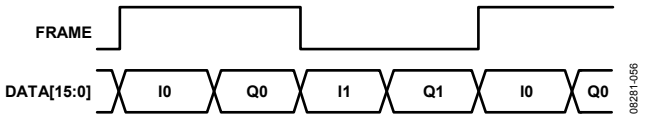

Figure 88. Timing Diagram of Extended FRAME Signal Required to Align Input Data for SED

The SED has three flag bits (Register 0x67, Bit 0, Bit 1, and Bit 5) that indicate the results of the input sample comparisons. The sample error detected bit (Register 0x67, Bit 5) is set when an error is detected and remains set until cleared. The SED also provides registers that indicate which input data bits experienced errors (Register 0x70 through Register 0x73). These bits are latched and indicate the accumulated errors detected until cleared.

The autoclear mode has two effects: it activates the compare fail bit and the compare pass bit (Register 0x67, Bit 1 and Bit 0) and changes the behavior of Register 0x70 through Register 0x73. The compare pass bit sets if the last comparison indicated the sample was error free. The compare fail bit sets if an error is detected. The compare fail bit is automatically cleared by the reception of eight consecutive error-free comparisons. When autoclear mode is enabled, Register 0x70 through Register 0x73 accumulate errors as previously described but reset to all 0s after eight consecutive error-free sample comparisons are made.

The sample error, compare pass, and compare fail flags can be configured to trigger an $\overline{\text{IRQ}}$ when active, if desired. This is done by enabling the appropriate bits in the event flag register (Register 0x07).

Table 27 shows a progression of the input sample comparison results and the corresponding states of the error flags.

SED EXAMPLE

Normal Operation

The following example illustrates the SED configuration for continuously monitoring the input data and assertion of an $\overline{\text{IRQ}}$ when a single error is detected.

1. Write to the following registers to enable the SED and load the comparison values:
   Register 0x67 → 0x80
   Register 0x68 → I0[7:0]
   Register 0x69 → I0[15:8]
   Register 0x6A → Q0[7:0]
   Register 0x6B → Q0[15:8]
   Register 0x6C → I1[7:0]
   Register 0x6D → I1[15:8]
   Register 0x6E → Q1[7:0]
   Register 0x6F → Q1[15:8]
   Comparison values can be chosen arbitrarily; however, choosing values that require frequent bit toggling provides the most robust test.
2. Enable the SED error detect flag to assert the $\overline{\text{IRQ}}$ pin.
   Register 0x05 → 0x04
3. Begin transmitting the input data pattern.

If $\overline{\text{IRQ}}$ is asserted, read Register 0x67 and Register 0x70 through Register 0x73 to verify that a SED error was detected and determine which input bits were in error. The bits in Register 0x70 through Register 0x73 are latched; therefore, the bits indicate any errors that occurred on those bits throughout the test and not just the errors that caused the error detected flag to be set.

Note that the FRAME signal is not required during normal operation when the device is configured for word mode. To enable the alignment of the I0 sample as described above requires the use of the FRAME signal. The timing diagram for byte and nibble modes are the same as during normal operation and are shown in Figure 44 and Figure 45, respectively.

Table 27. Progression of Comparison Outcomes and the Resulting SED Register Values

Compare Results (Pass/Fail)	P	F	F	F	P	P	P	P	P	P	P	P	P	F	P	F
Register 0x67, Bit 5 (Sample Error Detected)	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Register 0x67, Bit 1 (Compare Fail)	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1
Register 0x67, Bit 0 (Compare Pass)	1	0	0	0	1	1	1	1	1	1	1	1	1	0	1	0
Register 0x70 to Register 0x73 (Errors Detected x_BITS[15:0])	Z[1]	N[2]	N[2]	N[2]	N[2]	N[2]	N[2]	N[2]	N[2]	N[2]	N[2]	N[2]	Z[1]	N[2]	N[2]	N[2]

[1] Z = all 0s.
[2] N = nonzero.

EXAMPLE START-UP ROUTINE

There are certain sequences that should be followed to ensure reliable start-up of the AD9122. This section shows an example start-up routine assuming the configuration detailed in the following section.

Device Configuration

The following device configuration is used for this example:

f_{DATA} = 122.88MSPS
Interpolation = 4x, using HB1='10' and HB2='010010'
Input data = Baseband data
f_{OUT} = 140MHz
f_{REFCLK} = 122.88MHz
PLL = Enabled
Fine NCO = Enabled
Inverse SINC Filter = Enabled
Synchronization = Enabled
Silicon Revision = R2

Derived PLL Settings

The following PLL settings can be derived from the device configuration:

$f_{DACCLK}=f_{DATA}$*Interpolation=491.52MHz
f_{VCO} = 4*f_{DACCLK}=1966.08MHz (1GHz < f_{VCO} < 2GHz)
N1=f_{DACCLK}/f_{REFCLK}=4
N2=f_{VCO}/f_{DACCLK}=4

Derived NCO Settings

The following NCO settings can be derived from the device configuration:

f_{NCO} = 2 * f_{DATA}
$f_{CARRIER}=f_{OUT}-f_{MODHB1}$=140-122.88=17.12MHz
FTW=17.12/(2*122.88)*2^32=0x11D55555

Start-Up Sequence

The following sequences the power clock and register write sequencing for reliable device start-up:

```
Power up Device (no specific power supply sequence is required)
Apply stable REFCLK input signal.
Apply stable DCI input signal.
Issue H/W RESET (Optional)

Device Configuration Register Write Sequence:

0x00 → 0x20   /* Issue Software Reset */
0x00 → 0x00

0x0B → 0x20   /* Start PLL */
0x0C → 0xE1
0x0D → 0xD9
0x0A → 0xCF
0x0A → 0xA0
              /* ??Verify PLL is Locked?? */

Read 0x0E, Expect bit 7 = 0, bit 6 = 1
Read 0x06, Expect 0x5C

0x10 → 0x48   /* Choose Data Rate Mode */
0x17 → 0x04   /* Issue Software FIFO Reset */
0x18 → 0x02
0x18 → 0x00
              /* ??Verify FIFO Reset?? */

Read 0x18, Expect 0x05
Read 0x19, Expect 0x07

0x1B → 0x84   /* Configure Interpolation Filters */
0x1C → 0x04
0x1D → 0x24

0x1E → 0x01   /* Configure NCO */
0x30 → 0x55
0x31 → 0x55
0x32 → 0xD5
0x33 → 0x11
0x36 → 0x01   /* Update Frequency Tuning Word */
0x36 → 0x00
```

OUTLINE DIMENSIONS

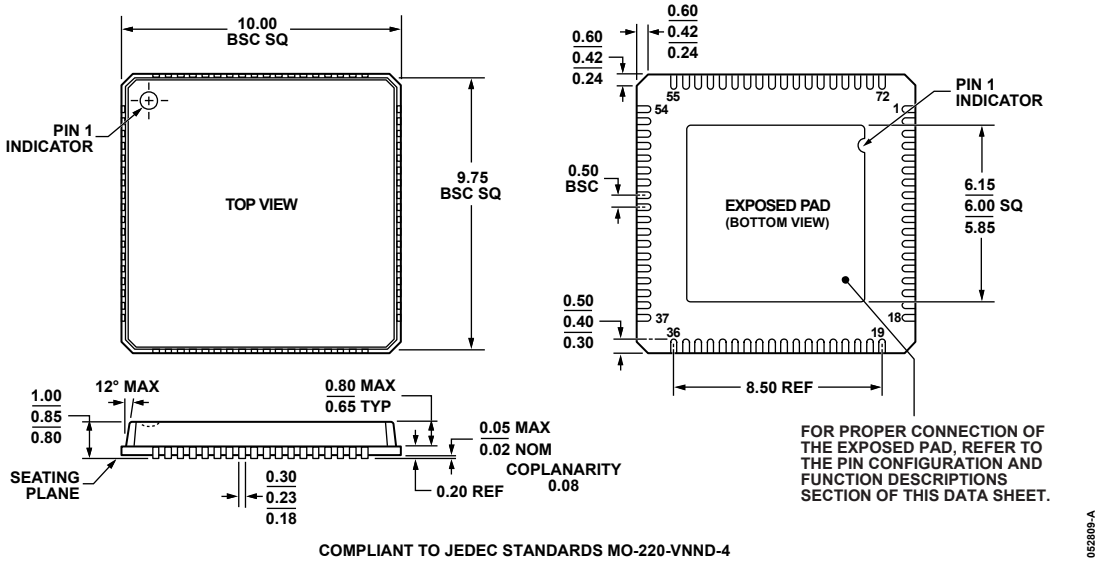

Figure 89. 72-Lead Lead Frame Chip Scale Package [LFCSP_VQ]
10 mm × 10 mm Body, Very Thin Quad
(CP-72-7)
Dimensions shown in millimeters

ORDERING GUIDE

Model[1]	Temperature Range	Package Description	Package Option
AD9122BCPZ	−40°C to +85°C	72-lead LFCSP_VQ	CP-72-7
AD9122BCPZRL	−40°C to +85°C	72-lead LFCSP_VQ	CP-72-7
AD9122-M5372-EBZ		Evaluation Board Connected to ADL5372 Modulator	
AD9122-M5375-EBZ		Evaluation Board Connected to ADL5375 Modulator	

[1]Z = RoHS Compliant Part.